

Studies in Computational Intelligence 693

Hocine Cherifi
Sabrina Gaito
Walter Quattrociocchi
Alessandra Sala *Editors*

Complex Networks & Their Applications V

Proceedings of the 5th International
Workshop on Complex Networks and
their Applications (COMPLEX NETWORKS
2016)

 Springer

Studies in Computational Intelligence

Volume 693

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series “Studies in Computational Intelligence” (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the worldwide distribution, which enable both wide and rapid dissemination of research output.

More information about this series at <http://www.springer.com/series/7092>

Hocine Cherifi · Sabrina Gaito
Walter Quattrociocchi · Alessandra Sala
Editors

Complex Networks & Their Applications V

Proceedings of the 5th International
Workshop on Complex Networks and their
Applications (COMPLEX NETWORKS
2016)

Editors

Hocine Cherifi
University of Burgundy
Dijon
France

Walter Quattrociocchi
IMT Lucca
Lucca
Italy

Sabrina Gaito
Computer Science Department
University of Milan
Milan
Italy

Alessandra Sala
Blanchardstown Business and Technology
Park
Bell Labs-Nokia
Blanchardstown
Ireland

ISSN 1860-949X

ISSN 1860-9503 (electronic)

Studies in Computational Intelligence

ISBN 978-3-319-50900-6

ISBN 978-3-319-50901-3 (eBook)

DOI 10.1007/978-3-319-50901-3

Library of Congress Control Number: 2016959260

© Springer International Publishing AG 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

This Springer imprint is published by Springer Nature

The registered company is Springer International Publishing AG

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The International Workshop on Complex Networks & their Applications was first held in 2012. It was initially conceived as a forum to bring together researchers from a wide variety of fields ranging from Computational Social Science, to Economic Complexity, up to Bioinformatics to review current scientific work and formulate new directions in network science. The tradition has continued with an annual single-track meeting that has become one of the leading international events in the field. Fuelled by the skills and expertise of participants from these diverse research fields, this workshop allows for cross-fertilization between fundamental and applied research. It offers a unique opportunity for reflection on the current state of the field, unanswered but critical questions, and potential future directions.

This volume of proceedings provides an opportunity for readers to engage with a selection of papers presented during the Fifth edition, hosted by the University of Milan (Italy), from November 30 to December 02, 2016. Although, they do not provide a fully comprehensive coverage of the field, the 65 papers selected by the Scientific Committee reflect the interdisciplinary nature of the scientific areas covered by the workshop. They have been organized in 11 sections reflecting multiple aspects of complex network research:

- Network models
- Network measures
- Community structure
- Network dynamics
- Diffusion, epidemics and spreading processes
- Resilience and control
- Network visualization
- Social and political networks
- Networks in finance and economics
- Biological and ecological networks
- Network analysis

A very encouraging response has been received by COMPLEX NETWORKS 2016 in terms of submissions. The 204 contributions that we received from 47 countries

around the world reflect the great vitality and diversity of the complex network community. All the submissions have been peer reviewed from at least 3 independent reviewers from our strong international program committee in order to ensure high quality of contributed material as well as adherence to the conference topics. After the review process, 65 papers were selected to be included in the proceedings.

Each edition of the workshop represents a challenge that cannot be successfully achieved without the deep involvement of numerous people and institutions. We address sincere thanks to all of them for their support, and to the University of Milan for making us so welcome.

We are very grateful to our keynote speakers for their plenary lectures covering different areas of the conference. The talk of Guido Caldarelli (IMT Lucca - Italy) focused on the origins of instability in financial networks. The presentation given by Raissa D'Souza (U. C. Davis - USA) dealt with the steering and controlling systems of interdependent networks. Renaud Lambiotte (University of Namur -Belgium) gave a talk on "Burstiness and spreading on networks: models and predictions" and Yamir Moreno (University of Zaragoza - Spain) presented the talk "On the structure and dynamics of multilayer networks". The talk given by Eiko Yoneki (University of Cambridge - UK) was about "Efficient large-scale graph processing" and Ben Y. Zhao (U. C. Santa-Barbara - USA) covered the link prediction issue from an empirical perspective. Their support of the workshop is without a doubt one of the reasons of the success of COMPLEX NETWORKS 2016.

Two speakers gave very illuminating tutorials that drew many conference participants. These talks, held on November 29, 2016 were accessible to a general audience of graduate students. Ernesto Estrada (University of Strathclyde Glasgow - UK) gave a lecture on "Consensus dynamics on networks. Theory and applications" and Bruno Gonçalves (New York University - USA) delivered a practical introduction to machine learning (with Python).

We record our thanks to our fellow members of the Organizing Committee: Chantal Cherifi (University of Lyon2 - France) and Antonio Scala (CNR - Italy), the poster chairs, for arranging the poster session program and the editing of the book of abstracts; Bruno Gonçalves (New York University - USA), the publicity chair, for his work in securing a substantial input of papers from both Asia and America and in encouraging participation from those areas; and all the session chairs for their outstanding participation. We would also like to record our appreciation for the work of the Local Arrangement Committee. In particular, Carlo Piccardi (Politecnico di Milano - Italy) and Fabio Della Rossa (Politecnico di Milano - Italy) in making all the excellent logistical arrangements for the conference. We also acknowledge the important contributions of the members of the Computer Science Department of the University of Milan. In particular, the team of the NPTLab (University of Milan) led by Gian Paolo Rossi. We thank him for his unwavering support. Many thanks to its junior members, Matteo Zignani and Christan Quadri for the incredible work they have done in the organization and the editing of the proceeding. We extend our thanks to Matteo Re and Giorgio Valentini, their efforts made a great contribution to the success of the workshop.

We are also indebted to our partners, Alessandro Fellegara and Alessandro Egro along with their team (Tribe Communication) for their passion and patience in designing the visual identity of the workshop. Our gratitude must also be extended to our sponsors, Blogmeter, Celi and Shaman, for supporting the workshop.

We would also like to express our deepest appreciation to all those who have helped us for the success of this meeting. Sincere thanks to the contributors, the success of the technical program would not be possible without their creativity. Finally, we would like to express our most sincere thanks to the Program Committee members who have so generously volunteered their precious time to support the peer review process.

We hope that this volume makes a useful contribution to issues surrounding the fascinating world of complex networks and that you enjoy the papers as much as we enjoyed organizing the conference and putting this collection of papers together.

Milan,
November 2016

Hocine Cherifi
Sabrina Gaito
Walter Quattrociocchi
Alessandra Sala

Organization & Committees

General Chair	Hocine Cherifi, University of Burgundy, France
Program Co-Chairs	Sabrina Gaito, University of Milan, Italy Walter Quattrociocchi, IMT Institute for Advanced Studies, Italy Alessandra Sala, Bell Labs Dublin, Ireland
Poster Chairs	Chantal Cherifi, University of Lyon 2, France Antonio Scala, Institute for Complex Systems / Italian National Research Council, Italy
Publicity Chair	Bruno Gonçalves, New York University, USA
Local Chairs	Carlo Piccardi, Politecnico di Milano, Italy Sabrina Gaito, University of Milan, Italy
Local Committee	Fabio Della Rossa, Politecnico di Milano, Italy Christian Quadri, University of Milan, Italy Matteo Re, University of Milan, Italy Giorgio Valentini, University of Milan, Italy Matteo Zignani, University of Milan, Italy
Web Chair	Matteo Zignani, University of Milan, Italy
Submission Chairs	Christian Quadri, University of Milan, Italy Matteo Zignani, University of Milan, Italy

**Program
Committee**

Sebastian Ahnert, University of Cambridge, UK
Luca Maria Aiello, Yahoo Labs, Spain
Tatsuya Akutsu, Kyoto University, Japan
Reka Albert, Pennsylvania State University, USA
Antoine Allard, Universitat de Barcelona, Spain
Claudio Altafini, Linköping University, Sweden
Alvarez-Zuzek Lucila, IFIMAR-UNMdP, Argentina
Fred Amblard, IRIT - University Toulouse 1 Capitole, France
Claudio Angione, Teesside University, UK
Antonioni Alberto, Carlos III University of Madrid, Spain
Nuno Araujo, Universidade de Lisboa, Portugal
Valerio Arnaboldi, IIT-CNR, Italy
Martin Atzmueller, University of Kassel, Germany
Rodolfo Baggio, Bocconi University, Italy
James Bagrow, University of Vermont, USA
Nikita Basov, St Petersburg State University, Russia
Gareth Baxter, University of Aveiro, Portugal
Rosa M. Benito, Universidad Politecnica de Madrid, Spain
Ginestra Bianconi, Queen Mary University, UK
Javier Borge-Holthoefer, IN3-UOC, Barcelona, Spain
Stefan Bornholdt, University of Bremen, Germany
Dan Braha, NECSI, USA
Marcus Brede, University of Bristol, UK
Piotr Brodka, Wroclaw University of Technology, Poland
Javier M. Buldu, Universidad Rey Juan Carlos, Spain
Raffaella Burioni, Università di Parma, Italy
Vincenza Carchiolo, Università di Catania, Italy
Alessio Cardillo, EPFL, Switzerland
Ciro Cattuto, ISI Foundation, Italy
Nitesh V. Chawla, University of Notre Dame, USA
Guanrong Chen, City University of Hong Kong, Hong Kong
Kwang-Cheng Chen, National Taiwan University, Taiwan
Chantal Cherifi, Lyon 2 University, France
Richard Clegg, Imperial College, London, UK
Jack Cole, ARL, USA
Luciano Costa, Universidade de SaPaulo, Brasil
Regino Criado, Universidad Rey Juan Carlos, Spain
Mihai Cucuringu, UCLA, USA
Jean-Charles Delvenne, University of Louvain, Belgium
Fabrizio De Vico Fallani, INRIA, France
Jana Diesner, University of Illinois Urbana-Champaign, USA
Florian Dorfler, ETH Zurich, Switzerland
Jordi Duch, Universitat Rovira i Virgili, Spain
Mohammed El Hassouni, Rabat University, Morocco
Ernesto Estrada, University of Strathclyde, UK

Tim Evans, Imperial College London, UK
Mauro Faccin, University of Louvain, Belgium
Giorgio Fagiolo, Sant'Anna School of Advanced Studies, Italy
Alessandro Flammini, Indiana University, USA
Eric Fleury, ENS Lyon / INRIA, France
Mattia Frasca, University of Catania, Italy
Jose Manuel Galan, University of Burgos, Spain
Antonios Garas, ETH Zurich, Switzerland
Gourab Ghoshal, University of Rochester, USA
Silvia Giordano, University of Applied Sciences and
Arts of Southern Switzerland, Switzerland
James Gleeson, University of Limerick, Ireland
Kwang-Il Goh, Korea University, South Korea
Sergio Gomez, Universitat Rovira i Virgili, Spain
Bruno Gonçalves, New York University, USA
Steve Gregory, University of Bristol, UK
Hasan Guclu, University of Pittsburgh, USA
Jean-Loup Guillaume, University of La Rochelle, France
Roger Guimera, Universitat Rovira i Virgili, Spain
Mehmet Gunes, University of Nevada, Reno, USA
Aric Hagberg, Los Alamos National Laboratory, USA
Chris Hankin, Imperial College London, UK
Laurent Hebert-Dufresne, Santa Fe Institute, USA
Desmond Higham, University of Strathclyde, UK
Seok-Hee Hong, University of Sydney, Australia
Philipp Hoevel, TU Berlin, Germany
Ulrich Hoppe, University of Duisburg-Essen, Germany
Pan Hui, Hong Kong Univ. of Science and Technology, Hong Kong
Gerardo Iniguez, UNAM, Mexico
Marco Javarone, Università di Sassari, Italy
Hawoong Jeong, KAIST, South Korea
Marcus Kaiser, Newcastle University, UK
Rushed Kanawati, Université Paris 13, France
Marton Karsai, ENS Lyon / INRIA, France
Dror Kenett, Boston University, USA
Hyoungshick Kim, Sungkyunkwan University, South Korea
Maksim Kitsak, Northeastern University, USA
Mikko Kivela, Aalto University, Finland
Peter Klimek, University of Vienna, Austria
Jerome Kunegis, University of Koblenz-Landau, Germany
Ryszard Kutner, University of Warsaw, Poland
Renaud Lambiotte, University of Namur, Belgium
Christine Largeron, Lyon University, France
Matthieu Latapy, CNRS-Paris 6, France
Anna T. Lawniczak, University of Guelph, Canada
Eric Leclercq, University of Burgundy, France

Sang Hoon, Lee, KIAS, South Korea
 Yang-Yu Liu, Harvard University, USA
 Alessandro Lomi, USI, Switzerland
 Alessandro Longheu, DIEEI - University of Catania, Italy
 John C.S. Lui, Chinese University of Hong Kong, Hong Kong
 Pdraig MacCarron, University of Oxford, UK
 Matteo Magnani, Uppsala University, Sweden
 Clémence, Magnien, Université Paris 6, France
 Giuseppe Mangioni, University of Catania, Italy
 Michael Mas, ETH Zurich, Switzerland
 Naoki Masuda, University of Bristol, UK
 Petr Matous, University of Sydney, Australia
 Natarajan Meghanathan, Jackson State University, USA
 Sandro Meloni, University of Zaragoza, Spain
 Jose Mendes, Universidade de Aveiro, Portugal
 Ronaldo Menezes, Florida Institute of Technology, USA
 Henning Meyerhenke, Karlsruhe University of Technology, Germany
 Radoslaw Michalski, Wroclaw University of Technology, Poland
 Tijana Milenkovic, University of Notre Dame, USA
 Giovanna Miritello, Telefonica Research, Spain
 Bivas Mitra, IIT Kharagpur, India
 Suzy Moat, University of Warwick, UK
 Misael Mongiovi, CNR, Italy
 Yamir Moreno, Universidad de Zaragoza, Spain
 Igor Mozetic, Jozef Stefan Institute, Slovenia
 Tsuyoshi Murata, Tokyo Institute of Technology, Japan
 Andrea Omicini, University of Bologna, Italy
 Eilsa Omodei, Universitat Rovira i Virgili, Spain
 Gergely Palla, Eötvös University, Hungary
 Pietro Panzarasa, Queen Mary University of London, UK
 Fragkiskos Papadopoulos, Cyprus University of Technology, Cyprus
 Symeon Papadopoulos, Information Technologies Institute, Greece
 Michela Papandrea, University of Applied Sciences and
 Arts of Southern Switzerland, Switzerland
 Han Woo Park, YeungNam University, South Korea
 Juyong Park, KAIST, South Korea
 Andrea Passarella, CNR, Italy
 Tiago Peixoto, Universität Bremen, Germany
 Matjaz Perc, University of Maribor, Slovenia
 Nicola Perra, University of Greenwich, UK
 Giovanni Petri, ISI Foundation, Italy
 Chiara Poletto, Inserm, Paris, France
 Marco Quaggiotto, ISI Foundation, Italy
 Asha Rao, RMIT University, Australia
 Massimo Riccaboni, IMT Lucca, Italy
 Luis E C Rocha, Karolinska Institutet, Sweden
 Luis M. Rocha, Indiana University, USA

Francisco Rodrigues, University of Sao Paulo, Brasil
Henrik Ronellenfitsch, University of Pennsylvania, USA
Marta Sales-Pardo, Universitat Rovira i Virgili, Spain
Hiroki Sayama, Binghamton University, USA
Antonio Scala, Institute for Complex Systems / Italian
National Research Council, Italy
Maximilian Schich, University of Texas at Dallas, USA
Ingo Scholtes, ETH Zurich, Switzerland
Frank Schweitzer, ETH Zurich, Switzerland
Aneesh Sharma, Twitter Inc., USA
Rajesh Sharma, University of Bologna, Italy
Filippo Simini, University of Bristol, UK
Anurag Singh, NIT Delhi, India
Persebastian Skardal, Trinity College, UK
Chaoming Song, University of Miami, USA
Mauro Sozio, Telecom Paris Tech, France
Markus Strohmaier, University of Koblenz-Landau, Germany
Michael Szell, Northeastern University, USA
Bosiljka Tadic, Jozef Stefan Institute, Ljubljana, Slovenia
Andrea Tagarelli, University of Calabria, Italy
I-Hsien Ting, National University of Kaohsiung, Taiwan
Olivier Togni, University of Burgundy, France
Jan Treur, Vrije Universiteit Amsterdam, Netherlands
Liubov Tupikina, PIK, Potsdam, Germany
Stephen Uzzo, New York Hall of Science, USA
Piet Van Mieghem, Delft University of Technology, Netherlands
Balazs Vedres, Central European University, Hungary
Huijuan Wang, Delft University of Technology, Netherlands
Xiao-Fan Wang, Shanghai Jiao Tong University, China
Pinghui Wang, Xi'an Jiaotong University, China
Christo Wilson, Northeastern University, USA
Bin Wu, Beijing University of Posts and Telecommunications, China
Zi-Ke Zhang, Hangzhou Normal University, China
Matteo Zignani, University of Milan, Italy



Contents

Part I Network models

A Hypotheses-driven Bayesian Approach for Understanding Edge Formation in Attributed Multigraphs	3
Lisette Espín-Noboa, Florian Lemmerich, Markus Strohmaier and Philipp Singer	
Generating Scaled Replicas of Real-World Complex Networks	17
Christian L. Staudt, Michael Hamann, Ilya Safro, Alexander Gutfraind and Henning Meyerhenke	
Modeling of Data Communication Networks using Dynamic Complex Networks and its Performance Studies	29
Suchi Kumari and Anurag Singh	
Testing for the signature of policy in online communities	41
Alberto Cottica, Guy Melançon and Benjamin Renoust	
A Temporal-Causal Network Model for the Relation Between Religion and Human Empathy	55
Laila van Ments, Peter Roelofsma and Jan Treur	
Network-Oriented Modeling and Analysis of Dynamics Based on Adaptive Temporal-Causal Networks	69
Jan Treur	
What governs a language’s lexicon? Determining the organizing principles of phonological neighbourhood networks	83
Rory Turnbull and Sharon Peperkamp	
Dominance, Deference, and Hierarchy Formation in Wikipedia Edit-Networks	95
Jürgen Lerner and Alessandro Lomi	

Part II Network Measures

Identifying Influential Spreaders by Graph Sampling 111
 Nikos Salamanos, Elli Voudigari and Emmanuel J. Yannakoudakis

Influential Actors Detection Using Attractiveness Model in Social Media Networks 123
 Ziyaad Qasem, Marc Jansen, Tobias Hecking and H.Ulrich Hoppe

Analyzing Multiple Rankings of Influential Nodes in Multiplex Networks 135
 Sude Tavassoli and Katharina A. Zweig

Preserving Sparsity in Dynamic Network Computations 147
 Francesca Arrigo and Desmond J. Higham

Flows of Knowledge in Citation Networks 159
 Benjamin Renoust, Vivek Claver and Jean-François Baffier

Detecting Nestedness in Graphs 171
 Alexander Grimm and Claudio J. Tessone

Clustering of Paths in Complex Networks 183
 Mareike Bockholt and Katharina A. Zweig

Complexity Analysis of “Small-World Networks” and Spanning Tree Entropy 197
 Raihana Mokhlissi, Dounia Lotfi, Joyati Debnath and Mohamed El Marraki

Graph Structure Similarity using Spectral Graph Theory 209
 Brian Crawford, Raluca Gera, Jeffrey House, Thomas Knuth and Ryan Miller

A genetic algorithm-based approach to mapping the diversity of networks sharing a given degree distribution and global clustering 223
 Peter Overbury, Istvan Z. Kiss and Luc Berthouze

Within network learning on big graphs using secondary memory-based random walk kernels 235
 Jianyi Lin, Marco Mesiti, Matteo Re and Giorgio Valentini

A Method for Evaluating the Navigability of Recommendation Algorithms 247
 Daniel Lamprecht, Markus Strohmaier and Denis Helic

Part III Community Structure

A New Decision Technique For Sub-community And Multi-Level Knowledge Extraction In Social Networks 263
 Joseph Ndong and Ibrahima Gueye

Vertex-centred Method to Detect Communities in Evolving Networks 275
 Maël Canu, Marie-Jeanne Lesot and Adrien Revault d’Allonnes

Clustering, Prominence and Social Network Analysis on Incomplete Networks 287
 Kshiteesh Hegde, Malik Magdon-Ismaïl, Boleslaw Szymanski and Konstantin Kuzmin

Evaluating the community partition quality of a network with a genetic programming approach 299
 Marco Buzzanca, Vincenza Carchiolo, Alessandro Longheu, Michele Malgeri and Giuseppe Mangioni

A graph-based meta-approach for tag recommendation 309
 Manel Hmimida and Rushed Kanawati

Community detection in visibility networks: an approach to categorize percussive influence on audio musical signals. 321
 Dirceu de Freitas Piedade Melo, Inacio de Sousa Fadigas and Hernane Borges de Barros Pereira

Can we recognize the next user’s mobile community? 335
 Ahlem Drif, Abdellah Boukerram, Yacine Slimani and Silvia Giordano

Part IV Dynamics on Networks

Why Amicus Curiae Cosigners Come and Go: A Dynamic Model of Interest Group Networks 349
 Dino P. Christenson and Janet M. Box-Steffensmeier

Contradictory information flow in networks with trust and distrust 361
 Giuseppe Primiero, Michele Bottone, Franco Raimondi and Jacopo Tagliabue

The Echo Chamber Effect in Twitter: does community polarization increase? 373
 Siying Du and Steve Gregory

Semantic Stability in Wikipedia 385
 Darko Stanisavljevic, Ilire Hasani-Mavriqi, Elisabeth Lex, Markus Strohmaier and Denis Helic

Coopetition and Cooperosity over Opinion Dynamics 397
 Domenico Tangredi, Raffaele Iervolino and Francesco Vasca

Effect of Direct Reciprocity on Continuing Prosperity of Social Networking Services 411
 Kengo Osaka, Fujio Toriumi and Toshiharu Sugawara

Co-evolution of two networks representing different social relations in NetSense 423
 Ashwin Bahulkar and Boleslaw K. Szymanski and Kevin Chan and Omar Lizardo

Part V Diffusion, Epidemics and Spreading Processes

The spread of ideas in a weighted threshold network 437
 Scott Cox, K.J. Horadam and Asha Rao

Information Diffusion in Heterogeneous Groups 449
 Jennifer M. Larson

A Novel Approach to Predict Retweets and Replies Based on Privacy and Complexity-Aware Feature Planes 459
 Kamini Garg, Valerio Arnaboldi and Silvia Giordano

Least Squares Method for Diffusion Source Localization in Complex Networks 473
 Mohammed Lalou and Hamamache Kheddouci

The effects of local network structure on disease spread in coupled networks 487
 W. Vermeer, B. Head and U. Wilensky

The Accuracy of Mean-Field Approximation for Susceptible-Infected-Susceptible Epidemic Spreading with Heterogeneous Infection Rates 499
 Bo Qu and Huijuan Wang

Die-out Probability in SIS Epidemic Processes on Networks 511
 Qiang Liu and Piet Van Mieghem

Part VI Resilience and Control

Robustness of Network Controllability to Degree-Based Edge Attacks . . . 525
 Jijju Thomas, Supratim Ghosh, Deven Parek, Derek Ruths and Justin Ruths

Use of Random Topics as Practical Control Signals in a Social Network Model 539
 Francesca Casamassima and Marco Cremonini

A Multiplex Approach to Urban Mobility 551
 A. Baggag, S. Abba, T. Zanouada, J. Borge-Holthoefer and J. Srivastava

Part VII Network Visualization

Efficient Genealogical Graph Layout 567
 Radek Marik

NodeTrix-Multiplex: Visual Analytics of Multiplex Small World Networks 579
 Shivam Agarwal, Amit Tomar and Jaya Sreevalsan-Nair

Part VIII Social and Political Networks

Structural Patterns of the Occupy Movement on Facebook 595
 Michela Del Vicario, Qian Zhang, Alessandro Bessi, Guido Caldarelli and Fabiana Zollo

Political Participation in Mexico through Twitter 607
 Julio César Amador Díaz López and C. A. Piña-García

Online election campaigning: Identifying political parties using likes and comments 619
 Jessica Liebig, Mohammad Adib Khairuddin and Asha Rao

Journalistic Relevance Classification in Social Network Messages: an Exploratory Approach 631
 Miguel Sandim, Paula Fortuna, Alvaro Figueira and Luciana Oliveira

Part IX Networks in Finance and Economics

Stock prices prediction via tensor decomposition and links forecast 645
 Alessandro Spelta

Who buys what, where: Reconstruction of the international trade flows by commodity and industry 657
 Yuichi Ikeda and Tsutomu Watanabe

Network of Networks: A Meta-model for Simulated Financial Markets . . 671
 Talal Alsulaiman and Khaldoun Khashanah

Part X Biological and Ecological Networks

Motif-Based Analysis of Effective Connectivity in Brain Networks 685
 J. Meier, M. Mörtens, A. Hillebrand, P. Tewarie and P. Van Mieghem

Functional Reconstruction of Dyadic and Triadic Subgraphs in Spiking Neural Networks 697
 Myles Akin, Alex Onderdonk, Rhonda Dzakpasu and Yixin Guo

Modeling and Extending Ecological Networks Using Land Similarity 709
 Gianni Fenu, Pier Luigi Pau and Danilo Dessì

Part XI Network Analysis

A graph-based, semi-supervised, credit card fraud detection system 721
 Bertrand Lebuchot, Fabian Braun, Olivier Caelen and Marco Saerens

Modeling City Locations as Complex Networks: An initial study 735
 Lu Zhou, Yang Zhang, Jun Pang and Cheng-Te Li

An analysis of the Bitcoin users graph: inferring unusual behaviours 749
 Damiano Di Francesco Maesa, Andrea Marino and Laura Ricci

Networks with Hierarchical Structure: Applications to the Patent Domain 761
 Nikolai Nefedov

Social Connection Dynamics in a Health Promotion Network 773
 Eric Fernandes de Mello Araújo, Michel Klein and Aart van Halteren

Social Networks and Construction of Culture: A Socio-Semantic Analysis of Art Groups 785
 Nikita Basov, Ju-Sung Lee and Artem Antoniuk

Water Supply Network Partitioning Based On Weighted Spectral Clustering 797
 Armando Di Nardo, Michele Di Natale, Carlo Giudicianni, Roberto Greco and Giovanni Francesco Santonastaso

Robust optimization of power network operation: storage devices and the role of forecast errors in renewable energies 809
 Carsten Matke, Daniel Bienstock, Gonzalo Muñoz, Shuoguang Yang, David Kleinhans and Sebastian Sager

An Image Segmentation Algorithm based on Community Detection 821
 Youssef Mourchid, Mohammed El Hassouni and Hocine Cherifi

Erratum to: Identifying Influential Spreaders by Graph Sampling E1

Author Index 831

Part I
Network models

A Hypotheses-driven Bayesian Approach for Understanding Edge Formation in Attributed Multigraphs

Lisette Espín-Noboa, Florian Lemmerich, Markus Strohmaier and Philipp Singer

Abstract Understanding edge formation represents a key question in network analysis. Various approaches have been postulated across disciplines ranging from network growth models to statistical (regression) methods. In this work, we extend this existing arsenal of methods with a hypotheses-driven Bayesian approach that allows to intuitively compare hypotheses about edge formation on attributed multigraphs. We model the multiplicity of edges using a simple categorical model and propose to express hypotheses as priors encoding our belief about parameters. Using Bayesian model comparison techniques, we compare the relative plausibility of hypotheses which might be motivated by previous theories about edge formation based on popularity or similarity. We demonstrate the utility of our approach on synthetic and empirical data. This work is relevant for researchers interested in studying mechanisms explaining edge formation in networks.

1 Introduction

Understanding edge formation in networks is a key interest of our research community. For example, social scientists are frequently interested in studying relations between entities within social networks, e.g., how social friendship ties form between actors and explain them based on attributes such as a person's gender, race, political affiliation or age in the network [18]. Similarly, the complex networks community suggests a set of generative network models aiming at explaining the formation of edges focusing on the two core principles of *popularity* and *similarity* [15]. Thus, a series of approaches to study edge formation have emerged including statistical (regression) tools [10, 23] and model-based approaches [6, 15, 24] specifically established in the physics and complex networks communities. Other disciplines such as the computer sciences, biomedical sciences or political sciences use these tools to

Lisette Espín-Noboa✉ · Florian Lemmerich✉ · Markus Strohmaier✉ · Philipp Singer✉
GESIS, University of Koblenz-Landau, e-mail: `firstname.(first)lastname@gesis.org`

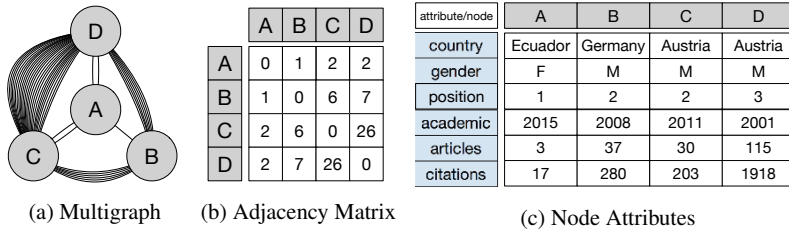


Fig. 1: **Example:** This example illustrates an unweighted attributed multigraph. (a) Shows a multigraph where nodes represent academic researchers, and edges scientific articles in which they have collaborated together. (b) Shows the adjacency matrix of the graph, where every cell represents the total number of edges between two nodes. (c) Decodes some attribute values per node. For instance, node D shows information about an *Austrian* researcher who started *his* academic career in *2001*.

answer empirical questions; e.g., co-authorship networks[12], wireless networks of biomedical sensors [19], or community structures of political blogs [1].

Problem Illustration. For illustration, consider Fig. 1; nodes represent authors, and (multiple) edges between them refer to co-authored scientific articles. Node attributes provide additional information about the authors, e.g., their home country and gender. An exemplary research question could be: “Can co-authorship be better explained by a mechanism that assumes more collaborations between authors from the *same country* or by a mechanism that assumes more collaborations between authors with the *same gender*?”. These and similar questions motivate the main objective of this work, which is to provide a Bayesian approach for understanding how edges emerge in networks based on some characteristics of the nodes.

While several methods for tackling such questions have been proposed, they come with certain limitations. For example, statistical regression methods based on QAP [5] or mixed-effects models [20] do not scale to large-scale data and results are difficult to interpret. For network growth models [15], it is necessary to find the appropriate model for a given hypothesis about edge formation and thus, it is often not trivial to intuitively compare competing hypotheses that sometimes might even go beyond simple popularity and similarity mechanisms. Consequently, we want to extend the methodological toolbox for studying edge formation in networks by proposing a first step towards a hypotheses-driven generative Bayesian framework.

Approach and methods. We focus on understanding edge formation in node-attributed multigraphs. We are interested in modeling and understanding the multiplicity of edges based on node attributes. Our approach follows a generative storyline. First, we define the model that can characterize the edge formation at interest. We focus on the simple categorical model, from which edges are independently drawn from. Motivated by previous work on sequential data [21], the core idea of our approach is to specify generative hypotheses about how edges emerge in a network. These hypotheses might be motivated by previous theories such as popularity or similarity [15]—e.g., for Fig. 1 we could hypothesize that authors are more likely to collaborate with each other if they are from the same country. Technically, we elicit these types of hypotheses as beliefs in parameters of the underlying categorical

model and encode and integrate them as priors into the Bayesian framework. Using Bayes factors with marginal likelihood estimations allows us to compare the relative plausibility of expressed hypotheses as they are specifically sensitive to the priors. The final output is a ranking of hypotheses based on their plausibility given the data.

Contributions. Our main contributions are: (i) We present a first step towards a Bayesian approach for comparing generative hypotheses about edge formation in networks. (ii) We provide simple categorical models based on local and global scenarios allowing the comparison of hypotheses for multigraphs. (iii) We provide guidelines for building hypotheses based on node attributes. (iv) We demonstrate the applicability of our approach on synthetic and empirical data. (v) We make an implementation of this approach openly available¹ on the Web.

2 Background

We start by introducing the underlying concepts of our approach.

Attributed Multigraphs. In this paper, we focus on *multigraphs* with *attributed nodes* and *unweighted edges without own identity*. That means, each pair of nodes can be connected by multiple indistinguishable edges, and there are features for the individual nodes available.

We formally define this as: Let $G = (V, E, F)$ be an unweighted attributed multigraph with $V = (v_1, \dots, v_n)$ being a list of nodes, $E = \{(v_i, v_j)\} \in V \times V$ a multiset of either directed or undirected edges, and a set of feature vectors $F = (f_1, \dots, f_n)$. Each feature vector $f_i = (f_i[1], \dots, f_i[c])^T$ maps a node v_i to c (numeric or categorical) attribute values. The graph structure is captured by an adjacency matrix $M_{n \times n} = (m_{ij})$, where m_{ij} is the multiplicity of edge (v_i, v_j) in E (i.e., number of edges between nodes v_i and v_j). By definition, the total number of multiedges is $l = |E| = \sum_{ij} m_{ij}$.

Fig. 1a shows an example unweighted attributed multigraph: nodes represent authors, and undirected edges represent co-authorship in scientific articles. The adjacency matrix of this graph—counting for multiplicity of edges—is shown in Fig. 1b. Feature vectors (node attributes) are described in Fig. 1c. Thus, for this particular case, we account for $n = 4$ nodes, $l = 44$ multiedges, and $c = 6$ attributes.

Bayesian Hypothesis Testing. Our approach compares hypotheses on edge formation based on techniques from Bayesian hypothesis testing [11, 21]. The elementary Bayes’ theorem states for parameters θ , given data D and a hypothesis H that:

$$\underbrace{P(\theta|D, H)}_{\text{posterior}} = \frac{\underbrace{P(D|\theta, H)}_{\text{likelihood}} \underbrace{P(\theta|H)}_{\text{prior}}}{\underbrace{P(D|H)}_{\text{marginal likelihood}}} \quad (1)$$

As observed data D , we use the adjacency matrix M , which encodes edges counts. θ refers to the model parameters, which in our scenario correspond to the probabilities of individual edges. H denotes a hypothesis under investigation. The *likelihood*

¹ <https://github.com/lisette-espin/JANUS>

	AA	AB	AC	AD	BB	BC	BD	CC	CD	DD	θ_{ij}
	$\frac{0}{44}$	$\frac{1}{44}$	$\frac{2}{44}$	$\frac{2}{44}$	$\frac{0}{44}$	$\frac{6}{44}$	$\frac{7}{44}$	$\frac{0}{44}$	$\frac{26}{44}$	$\frac{0}{44}$	

	A	B	C	D	θ_{ij}
A	$\frac{0}{5}$	$\frac{1}{5}$	$\frac{2}{5}$	$\frac{2}{5}$	θ_{1j}
B	$\frac{1}{14}$	$\frac{0}{14}$	$\frac{6}{14}$	$\frac{7}{14}$	θ_{2j}
C	$\frac{2}{34}$	$\frac{6}{34}$	$\frac{0}{34}$	$\frac{26}{34}$	θ_{3j}
D	$\frac{2}{35}$	$\frac{7}{35}$	$\frac{26}{35}$	$\frac{0}{35}$	θ_{4j}

(a) Global
(b) Local

Fig. 2: **Multigraph models:** This figure shows two ways of modeling the undirected multigraph shown in Fig. 1. That is, (a) global or graph-based model models the whole graph as a single distribution. (b) Local or neighbour-based model models each node as a separate distribution.

describes, how likely we observe data D given parameters θ and a hypothesis H . The *prior* is the distribution of parameters we believe in before seeing the data; in other words, the prior encodes our hypothesis H . The *posterior* represents an adjusted distribution of parameters after we observe D . Finally, the *marginal likelihood* (also called *evidence*) represents the probability of the data D given a hypothesis H .

In our approach, we exploit the sensitivity of the marginal likelihood on the prior to compare and rank different hypotheses: more plausible hypotheses imply higher evidence for data D . Formally, *Bayes Factors* can be employed for comparing two hypotheses. These are computed as the ratio between the respective marginal likelihood score. The strength of a Bayes factor can be judged using available interpretation tables [7]. While in many cases determining the marginal likelihood is computationally challenging and requires approximate solutions, we can rely on exact and fast-to-compute solutions in the models employed in this paper.

3 Approach

In this section, we describe the main steps towards a hypotheses-driven Bayesian approach for understanding edge formation in unweighted attributed multigraphs. To that end, we propose intuitive models for edge formation (Section 3.1), a flexible toolbox to formally specify belief in the model parameters (Section 3.2), a way of computing proper (Dirichlet) priors from these beliefs (Section 3.2), computation of the marginal likelihood in this scenario (Section 3.3), and guidelines on how to interpret the results (Section 3.4). We subsequently discuss these issues one-by-one.

3.1 Generative Edge Formation Models

We propose two variations of our approach, which employ two different types of generative edge formation models in multigraphs.

Global model. First, we utilize a simple *global model*, in which a fixed number of graph edges are randomly and independently drawn from the set of all potential edges in the graph G by sampling with replacement. Each edge (v_i, v_j) is sampled from

	A	B	C	D
A	0.0	0.1	0.1	0.1
B	0.1	0.0	0.1	0.1
C	0.1	0.1	0.0	0.9
D	0.1	0.1	0.9	0.0

	A	B	C	D
A	0.00	0.33	0.33	0.33
B	0.33	0.00	0.33	0.33
C	0.09	0.09	0.00	0.82
D	0.09	0.09	0.82	0.00

	A	B	C	D
A	1.00	2.33	2.33	2.33
B	2.33	1.00	2.33	2.33
C	1.36	1.36	1.00	4.27
D	1.36	1.36	4.27	1.00

(a) Belief matrix B_1 (b) Normalized B_1 (c) Prior $\kappa = 4$

Fig. 3: **Prior belief:** This figure illustrates the three main phases of prior elicitation. That is, (a) a matrix representation of belief B_1 , where authors are more likely to collaborate with each other if they are from the same country. (b) B_1 normalized row-wise using the local model interpretation. (c) Prior elicitation for $\kappa = 4$; i.e., $\alpha_{ij} = \frac{b_{ij}}{Z} \times \kappa + 1$.

a *categorical distribution* with parameters $\theta_{ij}, 1 \leq i \leq n, 1 \leq j \leq n, \forall ij : \sum_j \theta_{ij} = 1: (v_i, v_j) \sim \text{Categorical}(\theta_{ij})$. This means that each edge is associated with one probability θ_{ij} of being drawn next. Fig. 2a shows the maximum likelihood global model for the network shown in Fig. 1. Since this is an undirected graph, inverse edges can be ignored resulting in $n(n+1)/2$ potential edges/parameters.

Local models. As an alternative, we can also focus on a *local level*. Here, we model to which other node a specific node v will connect *given that any new edge starting from v is formed*. We implement this by using a set of n separate models for the outgoing edges of the ego-networks (i.e., the 1-hop neighborhood) of each of the n nodes. The ego-network model for node v_i is built by drawing randomly and independently a number of nodes v_j by sampling with replacement and adding an edge from v_i to this node. Each node v_j is sampled from a *categorical distribution* with parameters $\theta_{ij}, 1 \leq i \leq n, 1 \leq j \leq n, \forall i : \sum_j \theta_{ij} = 1: v_j \sim \text{Categorical}(\theta_{ij})$. The parameters θ_{ij} can be written as a matrix; the value in cell (i, j) specifies the probability that a new formed edge with source node v_i will have the destination node v_j . Thus, all values within one row always sum up to one. Local models can be applied for undirected and directed graphs (cf. also discussion in Section 6). In the directed case, we model only the outgoing edges of the ego-network. Fig. 2b depicts the maximum likelihood local models for our introductory example .

3.2 Hypothesis Elicitation

The main idea of our approach is to encode our beliefs in edge formation as Bayesian priors over the model parameters. As a common choice, we employ Dirichlet distributions as the *conjugate priors* of the categorical distribution. Thus, we assume that the model parameters θ are drawn from a Dirichlet distribution with hyperparameters $\alpha: \theta \sim \text{Dir}(\alpha)$. Similar to the model parameters themselves, the Dirichlet prior (or multiple priors for the local models) can be specified in a matrix. We will choose the parameters α in such a way that they reflect a specific belief about edge formation. For that purpose, we first specify matrices that formalize these beliefs, then we compute the Dirichlet parameters α from these beliefs.

Constructing Belief Matrices. We specify hypotheses about edge formation as *belief matrices* $B = b_{ij}$. These are $n \times n$ matrices, in which each cell $b_{ij} \in \mathbb{R}$ represents a belief of having an edge from node v_i to node v_j . To express a belief that an edge occurs more often (compared to other edges) we set b_{ij} to a higher value. In general, users have a large freedom to generate belief matrices. However, typical construction principles are to assume that nodes with specific attributes are more *popular* and thus edges connecting these attributes receive higher multiplicity, or to assume that nodes that are *similar* with respect to one or more attributes are more likely to form an edge, cf. [15]. Ideally, the elicitation of belief matrices is based on existing theories.

For example, based on the information shown in Fig. 1, one could “believe” that two authors collaborate *more frequently* together if: (1) they both are from the same country, (2) they share the same gender, (3) they have high positions, or (4) they are popular in terms of number of articles and citations. We capture each of these beliefs in one matrix. One implementation of the matrices for our example beliefs could be:

- B_1 (same country): $b_{ij} := 0.9$ if $f_i[\text{country}] = f_j[\text{country}]$ and 0.1 otherwise
- B_2 (same gender): $b_{ij} := 0.9$ if $f_i[\text{gender}] = f_j[\text{gender}]$ and 0.1 otherwise
- B_3 (hierarchy): $b_{ij} := f_i[\text{position}] \cdot f_j[\text{position}]$
- B_4 (popularity): $b_{ij} := f_i[\text{articles}] + f_j[\text{articles}] + f_i[\text{citations}] + f_j[\text{citations}]$

Fig. 3a shows the matrix representation of belief B_1 , and Fig. 3b its respective row-wise normalization for the local model case. While belief matrices are identically structured for local and global models, the ratio between parameters in different rows is crucial for the global model, but irrelevant for local ones.

Eliciting a Dirichlet prior. In order to obtain the hyperparameters α of a prior Dirichlet distribution we utilize the pseudo-count interpretation of the parameters α_{ij} of the Dirichlet distribution, i.e., a value of α_{ij} can be interpreted as $\alpha_{ij} - 1$ previous observations of the respective event for $\alpha_{ij} \geq 1$. We distribute pseudo-counts proportionally to a belief matrix. Consequently, the hyperparameters can be expressed as: $\alpha_{ij} = \frac{b_{ij}}{Z} \times \kappa + 1$, where κ is the concentration parameter of the prior. The normalization constant Z is computed as the sum of all entries of the belief matrix in the global model, and as the respective row sum in the local case. We suggest to set $\kappa = n \times k$, $k = \{0, 1, \dots, 10\}$. A high value of κ expresses a strong belief in the prior parameters. A similar alternative method to obtain Dirichlet priors is the *trial roulette method* [21]. For the global model variation, all α values are parameters for the same Dirichlet distribution, whereas in the local model variation, each row parametrizes a separate Dirichlet distribution.

3.3 Computation of the Marginal Likelihood

For comparing the relative plausibility of hypotheses we use the marginal likelihood. This is the aggregated likelihood over all possible values of the parameters θ weighted by the Dirichlet prior. For our set of local models we can calculate them as:

$$P(D|H) = \prod_{i=1}^n \frac{\Gamma(\sum_{j=1}^n \alpha_{ij})}{\Gamma(\sum_{j=1}^n \alpha_{ij} + m_{ij})} \prod_{j=1}^n \frac{\Gamma(\alpha_{ij} + m_{ij})}{\Gamma(\alpha_{ij})} \quad (2)$$

Recall, α_{ij} encodes our prior belief connecting nodes v_i and v_j in G , and m_{ij} are the actual edge counts. Since we evaluate only a single model in the global case, the product over rows i of the adjacency matrix can be removed, and we obtain:

$$P(D|H) = \frac{\Gamma(\sum_{i=1}^n \sum_{j=1}^n \alpha_{ij})}{\Gamma(\sum_{i=1}^n \sum_{j=1}^n \alpha_{ij} + m_{ij})} \prod_{i=1}^n \prod_{j=1}^n \frac{\Gamma(\alpha_{ij} + m_{ij})}{\Gamma(\alpha_{ij})} \quad (3)$$

Equation (3) holds for directed networks. In the undirected case, indices j go from i to n accounting for only half of the matrix including the diagonal. For a detailed derivation of the marginal likelihood given a Dirichlet-Categorical model see [22, 25]. For both models we focus on the log-marginal likelihoods in practice to avoid underflows.

Bayes Factor. Formally, we compare the relative plausibility of hypotheses by using so-called *Bayes factors* [7], which simply are the ratios of the marginal likelihoods for two hypotheses H_1 and H_2 . If it is positive, the first hypothesis is judged as more plausible. The strength of the Bayes factor can be checked in an interpretation table provided by Kass and Raftery [7].

3.4 Application of the Method and Interpretation of Results

We now showcase an example application of our approach featuring the network shown in Fig. 1, and demonstrate how results can be interpreted. For that purpose and due to space limitations, we focus on the local models variant.

Hypotheses. We compare four hypotheses (represented as belief matrices) B_1 , B_2 , B_3 , and B_4 elaborated in Section 3.2. Additionally, we use the *uniform hypothesis* as a *baseline*. It assumes that all edges are equally likely, i.e., $b_{ij} = 1$ for all i, j . Hypotheses that are not more plausible than the uniform cannot be assumed to capture relevant underlying mechanisms of edge formation. We also use the *data hypothesis* as an upper bound for comparison, which employs the observed adjacency matrix as belief: $b_{ij} = m_{ij}$.

Calculation and visualization. For each hypothesis H and every κ , we can elicit the Dirichlet priors (cf. Section 3.2), determine the aggregated marginal likelihood (cf. Section 3.3), and compare the plausibility of hypotheses compared to the uniform hypothesis at the same κ by calculating the logarithm of the Bayes factor as $\log(P(D|H)) - \log(P(D|H_{uniform}))$. We suggest two ways of visualizing the results, i.e., plotting the marginal likelihood values (Fig. 4a) or showing the Bayes factors (Fig. 4b) on the y-axis. In both cases, the x-axis refers to the concentration parameter κ . While the visualization showing directly the marginal likelihoods carries more information, visualizing Bayes factors makes it easier to spot smaller differences between the hypotheses.

Interpretation. Every line in both figures represents a hypothesis. In Fig. 4a, higher evidence values mean higher plausibility. Similarly, in Fig. 4b positive Bayes factors

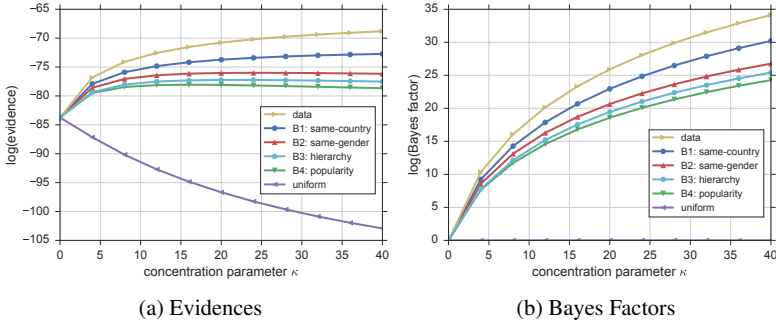


Fig. 4: **Ranking of hypotheses for the introductory example.** Rankings can be visualized using (a) the marginal likelihood or evidence (y-axis), or (b) Bayes factors (y-axis) by setting the uniform hypothesis as a baseline to compare with; higher values refer to higher plausibility. The x-axis depicts the concentration parameter κ . For this example, authors from the multigraph shown in Fig. 1 appear to prefer to collaborate more often with researchers of the same country rather than due to popularity (i.e., number of articles and citations). Note that all hypotheses outperform the uniform, meaning that they all represent reasonable explanations of edge formation for the given graph.

mean that for a given κ , the hypothesis is judged to be more plausible than the uniform baseline hypothesis; here, the relative Bayes factors also provide a ranking. If evidences or Bayes factors are increasing with κ , we can interpret this as further evidence for the plausibility of expressed hypothesis as this means that the more we believe in it, the higher the Bayesian approach judges its plausibility. As a result for our example, we see that the hypothesis believing that two authors are more likely to collaborate if they are from the same country is the most plausible one (after the data hypothesis). In this example, all hypotheses appear to be more plausible than the baseline, but this is not necessarily the case in all applications.

4 Experiments

We demonstrate the utility of our approach on both synthetic and empirical networks. Due to space limitations, we only showcase the local model results.

4.1 Synthetic Attributed Multigraph

We start with experiments on a synthetic attributed multigraph. Here, we control the underlying mechanisms of how edges in the network emerge and thus, expect these also to be good hypotheses for our approach.

Network. The network contains 100 nodes where each node is assigned one of two colors with uniform probability. For each node, we then randomly drew 200 undirected edges where each edge connects randomly with probability $p = 0.8$ to a

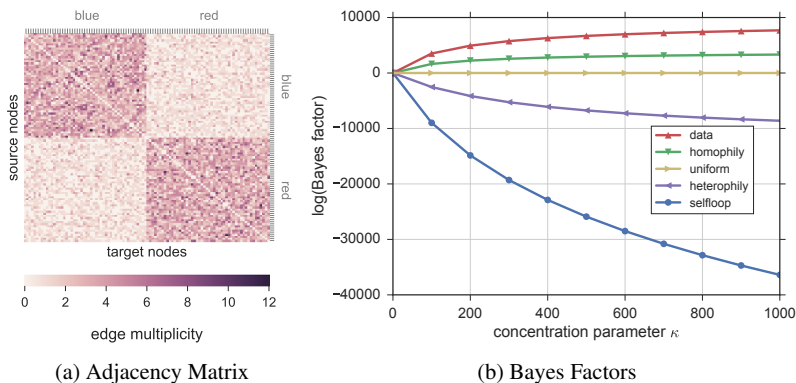


Fig. 5: **Ranking of hypotheses for synthetic network.** In (a), we show the adjacency matrix of the 2-color random multigraph with a node correlation of 80% for nodes of the same color and 20% otherwise. One can see homophily based on more connections between nodes of the same color; the diagonal is zero as there are no self-connections. In (b), we show the ranking of hypotheses based on Bayes factors when compared to the uniform hypothesis. As expected, the homophily hypothesis explains the edge formation best (positive Bayes factor), and the heterophily and selfloop hypotheses show negative Bayes factors—i.e., they provide no good explanations for edge formation.

different node of the same color, and with $p = 0.2$ to a node of the opposite color. The adjacency matrix of this graph is visualized in Fig. 5a.

Hypotheses. In addition to the uniform baseline hypothesis, we construct two intuitive hypotheses based on the node color that express belief in possible edge formation mechanics. First, the *homophily hypothesis* assumes that nodes of the same color are more likely to have more edges between them. Therefore, we arbitrary set belief values b_{ij} to 80 when nodes v_i and v_j are of the same color, and 20 otherwise. Second, the *heterophily hypothesis* expresses the opposite behavior; i.e., $b_{ij} = 80$ if the color of nodes v_i and v_j are different, and 20 otherwise. An additional *selfloop hypothesis* only believes in self-connections (i.e., diagonal of adjacency matrix).

Results. Fig. 5b shows the ranking of hypotheses based on their Bayes factors compared to the uniform hypothesis. Clearly, the homophily hypothesis is judged as the most plausible. This is expected and corroborates the fact that network connections are biased towards nodes of the same color. The heterophily and selfloop hypotheses show negative Bayes factors; thus, they are not good hypotheses about edge formation in this network. Due to the fact that the multigraph lacks of selfloops, the selfloop hypothesis decreases very quickly with increasing strength of belief κ .

4.2 Empirical Attributed Multigraph

Here, we focus on a real-world contact network based on wearable sensors.

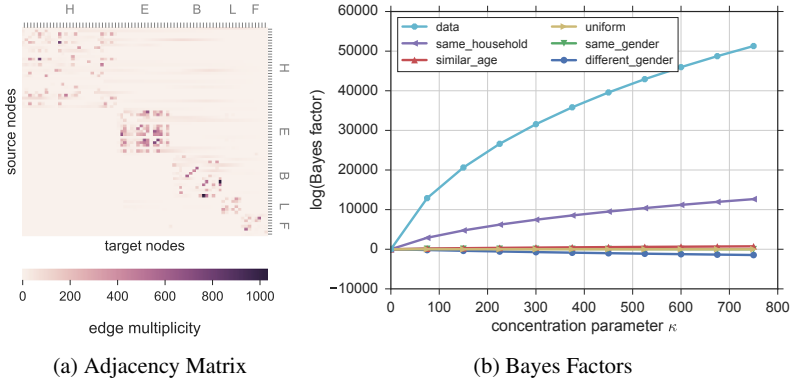


Fig. 6: **Ranking of hypotheses for Kenya contact network.** (a) Shows the adjacency matrix of the network with node ordering according to household membership. Darker cells indicate more contacts. (b) Displays the ranking of hypotheses based on Bayes factors, using the uniform hypothesis as baseline. The *same household hypothesis* (people are more likely to contact people from the same household) ranks highest. While the *similar age* hypothesis also provide positive Bayes Factors, the *same* and *different gender* hypotheses are less plausible than the baseline (uniform edge formation). Results are consistent for all κ .

Network. We study a network² capturing interactions of 5 households in rural Kenya between April 24 and May 12, 2012 [9]. The undirected unweighted multigraph contains 75 nodes (persons) and 32 643 multiedges (contacts) which we aim to explain. For each node, we know information such as gender and age (encoded into 5 age intervals). Interactions exist within and across households. Fig. 6a shows the adjacency matrix (i.e., number of contacts between two people) of the network. Household membership of nodes (rows/columns) is shown accordingly.

Hypotheses. We investigate edge formation by comparing—next to the uniform baseline hypothesis—four hypotheses based on node attributes as prior beliefs. (i) The *similar age hypothesis* expresses the belief that people of similar age are more likely to interact with each other. Entries b_{ij} of the belief matrix B are set to the inverse age distance between members: $\frac{1}{1+abs(f_i[age]-f_j[age])}$. (ii) The *same household hypothesis* believes that people are more likely to interact with people from the same household. We arbitrarily set b_{ij} to 80 if person v_i and person v_j belong to the same household, and 20 otherwise. (iii) With the *same gender hypothesis* we hypothesize that the number of same-gender interactions is higher than the different-gender interactions. Therefore, every entry b_{ij} of B is set to 80 if persons v_i and v_j are of the same gender, and 20 otherwise. Finally, (iv) the *different gender hypothesis* believes that it is more likely to find different-gender than same-gender interactions; b_{ij} is set to 80 if person v_i has the opposite gender of person v_j , and 20 otherwise.

Results. The results shown in Fig. 6b indicate that the *same household hypothesis* explains the data the best, since it has been ranked first and it is more plausible than the uniform. The *similar age* hypothesis also indicates plausibility due to positive

² <http://www.sociopatterns.org/datasets/kenyan-households-contact-network/>

Bayes factors. Both the *same* and *different gender hypotheses* show negative Bayes factors when compared to the uniform hypothesis suggesting that they are not good explanations of edge formation in this network. This gives us a better understanding of potential mechanisms producing underlying edges. People prefer to contact people from the same household and similar age, but not based on gender preferences. Additional experiments could further refine these hypotheses (e.g., combining them).

5 Related Work

We provide a broad overview of research on modeling and understanding edge formation in networks; i.e., *edge formation models* and *hypothesis testing on networks*.

Edge formation models. A variety of models explaining underlying mechanisms of *network formation* have been proposed. Here, we focus on models explaining linkage between dyads beyond structure by incorporating node attribute information. Prominently, the *stochastic blockmodel* [6] aims at producing and explaining communities by accounting for node correlation based on attributes. The *attributed graph* [16] models network structure and node attributes by learning the attribute correlations in the observed network. Furthermore, the *multiplicative attributed graph* [8] takes into account attribute information from nodes to model network structure. This model defines the probability of an edge as the product of individual attribute link formation affinities. *Exponential random graph models* [17] (also called the p^* class of models) represent graph distributions with an exponential linear model that uses feature-structure counts such as reciprocity, k-stars and k-paths. In this line of research, *p1 models* [4] consider expansiveness (sender) and popularity (receiver) as fixed effects associated with unique nodes in the network [3], in contrast to the *p2 models* [17] which account for random effects and assume dyadic independence conditionally to node-level attributes. While many of these works focus on binary relationships, [27] proposes an unsupervised model to estimate continuous-valued relationship strength for links from interaction activity and user similarity in social networks.

Hypothesis testing on networks. Previous works have implemented different techniques to test hypotheses about network structure. For instance, the work in [13] proposes an algorithm to determine whether two observed networks are significantly different. Another branch of research has specifically focused on dyadic relationships utilizing regression methods accounting for interdependencies in network data. Here, we find the state-of-the-art *Multiple Regression Quadratic Assignment Procedure* (MRQAP) [10] and its predecessor QAP [5] which permute nodes in such a way that the network structure is kept intact; this allows to test for significance of effects. *Mixed-effects models* [20] add random effects to the models allowing for variation to mitigate non-independence between responses (edges) from the same subject (nodes) [26]. Based on the *quasi essential graph* the work in [14] proposes to compare two graphs (i.e., Bayesian networks) by testing and comparing multiple hypotheses on their edges. Recently, the *generalized hypergeometric ensembles* [2] have been proposed as a framework for model selection and statistical hypothesis testing of finite, directed and weighted networks that allow to encode several topological patterns

such as block models where homophily plays an important role in linkage decision. In contrast to our work, neither of these approaches is based on Bayesian hypothesis testing, which avoids some fundamental issues of classic frequentist statistics.

6 Discussion

Next, we discuss some aspects and open questions related to the proposed approach.

Inconsistency of local model. For directed networks, the local ego-network models can assemble a full graph model by defining a probability distribution for the degrees of the source nodes of edges. For undirected networks, this is not directly possible as e.g., the ego-network model for v_A generated an edge from v_A to v_B , but the ego-network model for node v_B did not generate any edge to v_A . Note that this does not affect our comparison of hypotheses as we characterize the network.

Sparse data-connections. Most real networks exhibit small world properties such as high clustering coefficient and fat-tailed degree distributions meaning that the adjacency matrices are sparse. While comparison still relatively judges the plausibility, our hypotheses do not approximate the data curve as shown in Fig. 6b. As an alternative, one might want to limit our beliefs to only those edges that exist in the network, i.e., we would then only build hypotheses on how edge multiplicity varies between edges. Ultimately, our models also warrant extensions to adhere to the degree sequence in the network, e.g., in the direction of multivariate hypergeometric distributions as recently proposed in [2].

Other limitations and future work. The main intent of this work is the introduction of a hypotheses-driven Bayesian approach for understanding edge formation in networks. To that end, we showcased this approach on simple categorical models that warrant extensions, e.g., by incorporating appropriate models for other types of networks such as weighted or temporal networks. We can further investigate how to build good hypotheses by leveraging all node attributes, and infer subnetworks that fit best each of the given hypotheses. Moreover, there can be alternatives for non-attributed networks. For instance, one could use other networks (same nodes, different connections) to verify whether edges from a specific network can be explained by the mechanisms of other networks. In the future, we also plan an extensive comparison to other methods such as MRQAP, mixed-effects models and p^* models.

7 Conclusions

In this paper, we have presented a Bayesian framework that facilitates the understanding of edge formation in attributed multigraphs. The main idea is based on expressing hypotheses as beliefs in parameters (i.e., multiplicity of edges), incorporate them as priors, and utilize Bayes factors for comparing their plausibility. We proposed simple local and global Dirichlet-categorical models and showcased their utility on synthetic and empirical data. For illustration purposes our examples are based on small networks. We tested our approach with larger networks obtaining identical

results. In future, our concepts can be extended to further models such as models adhering to fixed degree sequences. We hope that our work contributes new ideas to the research line of understanding edge formation in complex networks.

Acknowledgements This work was partially funded by DFG German Science Fund research projects “KonSKOE” and “PoSTs II”.

References

- [1] Adamic, L.A., Glance, N.: The political blogosphere and the 2004 us election: divided they blog. In: Proceedings of the 3rd int. workshop on Link discovery, pp. 36–43. ACM (2005)
- [2] Casiraghi, G., Nanumyan, V., Scholtes, I., Schweitzer, F.: Generalized hypergeometric ensembles: Statistical hypothesis testing in complex networks. arXiv:1607.02441 (2016)
- [3] Goldenberg, A., Zheng, A.X., Fienberg, S.E., Airoldi, E.M.: A survey of statistical network models. *Foundations and Trends® in Machine Learning* **2**(2), 129–233 (2010)
- [4] Holland, P.W., Leinhardt, S.: An exponential family of probability distributions for directed graphs. *Journal of the American Statistical Association* **76**(373), 33–50 (1981)
- [5] Hubert, L., Schultz, J.: Quadratic assignment as a general data analysis strategy. *British journal of mathematical and statistical psychology* **29**(2), 190–241 (1976)
- [6] Karrer, B., Newman, M.E.: Stochastic blockmodels and community structure in networks. *Physical Review E* **83**(1), 016,107 (2011)
- [7] Kass, R.E., Raftery, A.E.: Bayes factors. *Journal of the American Statistical Association* **90**(430), 773–795 (1995)
- [8] Kim, M., Leskovec, J.: Modeling social networks with node attributes using the multiplicative attribute graph model. In: UAI 2011, Barcelona, Spain, July 14–17, 2011, pp. 400–409 (2011)
- [9] Kiti, M.C., Tizzoni, M., Kinyanjui, T.M., Koech, D.C., Munywoki, P.K., Meriac, M., Cappa, L., Panisson, A., Barrat, A., Cattuto, C., et al.: Quantifying social contacts in a household setting of rural kenya using wearable proximity sensors. *EPJ Data Science* **5**(1), 1 (2016)
- [10] Krackhardt, D.: Predicting with networks: Nonparametric multiple regression analysis of dyadic data. *Social networks* **10**(4), 359–381 (1988)
- [11] Kruschke, J.: *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press (2014)
- [12] Martin, T., Ball, B., Karrer, B., Newman, M.: Coauthorship and citation patterns in the physical review. *Physical Review E* **88**(1), 012,814 (2013)
- [13] Moreno, S., Neville, J.: Network hypothesis testing using mixed kronecker product graph models. In: *Data Mining (ICDM)*, pp. 1163–1168. IEEE (2013)
- [14] Nguyen, H.T.: Multiple hypothesis testing on edges of graph: a case study of bayesian networks
- [15] Papadopoulos, F., Kitsak, M., Serrano, M.Á., Boguná, M., Krioukov, D.: Popularity versus similarity in growing networks. *Nature* **489**(7417), 537–540 (2012)
- [16] Pfeiffer III, J.J., Moreno, S., La Fond, T., Neville, J., Gallagher, B.: Attributed graph models: Modeling network structure with correlated attributes. In: *WWW*, pp. 831–842. ACM (2014)
- [17] Robins, G., Pattison, P., Kalish, Y., Lusher, D.: An introduction to exponential random graph (p*) models for social networks. *Social networks* **29**(2), 173–191 (2007)
- [18] Sampson, S.F.: *A novice in a period of change: An experimental and case study of social relationships*. Cornell University (1968)
- [19] Schwiebert, L., Gupta, S.K., Weinmann, J.: Research challenges in wireless networks of biomedical sensors. In: *Proceedings of the 7th annual international conference on Mobile computing and networking*, pp. 151–165. ACM (2001)
- [20] Shah, K.R., Sinha, B.K.: *Mixed Effects Models*, pp. 85–96. Springer New York (1989)
- [21] Singer, P., Helic, D., Hotho, A., Strohmaier, M.: Hyptrails: A bayesian approach for comparing hypotheses about human trails on the web. *WWW*, pp. 1003–1013. ACM (2015)

- [22] Singer, P., Helic, D., Taraghi, B., Strohmaier, M.: Detecting memory and structure in human navigation patterns using markov chain models of varying order. *PLoS one* **9**(7), e102,070 (2014)
- [23] Snijders, T., Sreen, M., Zwaagstra, R.: The use of multilevel modeling for analysing personal networks: Networks of cocaine users in an urban area. *Journal of quantitative anthropology* **5**(2), 85–105 (1995)
- [24] Snijders, T.A.: Statistical models for social networks. *Review of Sociology* **37**, 131–153 (2011)
- [25] Tu, S.: The dirichlet-multinomial and dirichlet-categorical models for bayesian inference. Computer Science Division, UC Berkeley (2014)
- [26] Winter, B.: Linear models and linear mixed effects models in r with linguistic applications. [arXiv:1308.5499](https://arxiv.org/abs/1308.5499) (2013)
- [27] Xiang, R., Neville, J., Rogati, M.: Modeling relationship strength in online social networks. In: WWW, pp. 981–990. ACM (2010)

Generating Scaled Replicas of Real-World Complex Networks

Christian L. Staudt, Michael Hamann, Ilya Safro, Alexander Gutfraind and Henning Meyerhenke

Abstract Research on generative models plays a central role in the emerging field of network science, studying how statistical patterns found in real networks can be generated by formal rules. During the last two decades, a variety of models has been proposed with an ultimate goal of achieving comprehensive realism for the generated networks. In this study, we (a) introduce a new generator, termed ReCoN; (b) explore how models can be fitted to an original network to produce a structurally similar replica, and (c) aim for producing much larger networks than the original exemplar. In a comparative experimental study, we find ReCoN often superior to many other state-of-the-art network generation methods. Our design yields a scalable and effective tool for replicating a given network while preserving important properties at both micro- and macroscopic scales and (optionally) scaling the replica by orders of magnitude in size. We recommend ReCoN as a general practical method for creating realistic test data for the engineering of computational methods on networks, verification, and simulation studies. We provide scalable open-source implementations of most studied methods, including ReCoN.

1 Introduction

Context. When engineering algorithms, the ability to create good synthetic test data sets is valuable to estimate effectiveness and scalability of the proposed methods. A shortage of real data for this purpose can for example arise if they are proprietary,

Christian L. Staudt (e-mail: christian.staudt@kit.edu)✉ · Michael Hamann (e-mail: michael.hamann@kit.edu)✉ · Henning Meyerhenke (e-mail: meyerhenke@kit.edu)✉

Institute of Theoretical Informatics, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Ilya Safro (e-mail: isafro@clemson.edu)✉

School of Computing, Clemson University, Clemson, SC, USA

Alexander Gutfraind (e-mail: agutfraind.research@gmail.com)✉

Loyola University Medical Center, Maywood, IL, USA/Uptake, Inc., Chicago, IL, USA;

sensitive, or unavailable in different scales. In the context of developing network analysis algorithms, realistic synthetic graphs allow us to produce experimental results that are representative for what can be observed for real data. Among the main use cases are *obfuscation* (replacing restricted real data with similar synthetic data), *compression* (storing only a generator and its parameters instead of large graphs), as well as *extrapolation and sampling* (generating data at larger or smaller scales).

Problem definition. We envision two usage scenarios: Given an original (or real) network $O = (V, E)$ ($n_o = |V|$, and $m_o = |E|$) that cannot be freely shared, we would like to be able to create a synthetic network R (with n_r nodes) that matches the original in essential structural properties, so that computational results obtained from processing this network are representative for what the original network would yield. We refer to R as a *replica*. We assume that whoever creates the replica has access to O and can pass it to a *model fitting* algorithm which uses it to parametrize a generative model.

More importantly, in addition to producing *scale-1 replicas* (where $n_r = n_o$), in the second scenario we want to use the generative model for *extrapolation*: We want to parametrize it so that it produces a *scaled replica* R^x that has $n_r = x \cdot n_o$ nodes, where x is called the *scaling factor*. The structural properties of R^x should be such that they resemble a later growth stage of the original (also see Sec. 2). This should enable users of the replica to extrapolate the behavior of their methods when the network data is significantly scaled.

Finally, with respect to performance, we would like the generator algorithm and implementation as well as the fitting scheme to be efficient enough to produce large data sets (on the order of several millions of nodes and edges) quickly in practice.

State of the art. Many generative models for complex networks exist. We point the interested reader to a survey [12] for a more comprehensive overview. A widely used model intended for model fitting uses exponential random graph models (ERGM), cf. e. g. [25]. Unfortunately, ERGM are so expensive that graphs with tens of thousands of nodes are already considered big for these models [3].

Other generative models admit fast generators and are thus in our focus. Among those models are RMAT [6], BTER [16], and Hyperbolic Unit Disk Graphs (HUDG) [17]. Initially, they can fit only few properties of the original network by design, though. A previous fitting scheme by Leskovec et al. [20] for RMAT graphs is quite time-consuming already for medium-sized networks [28, 29].

Editing models create a synthetic network by editing the original network. The MUSKETEER generator [14] implements a multiscale editing model and is effective for obfuscation purposes. However, its current implementation [13] is not fast enough to generate sufficiently scaled replicas of large graphs.

Outline and contribution. In this paper we develop and evaluate a sufficiently fast generator that focuses on creating realistic *scaled* replicas of complex networks.

We point out in Section 2 which criteria we consider important for calling a (scaled) replica realistic. In particular we conceptualize realism in two ways: (i) matching an original graph in a set of important structural properties, and (ii) matching the running time behavior of various graph algorithms.

Our new generator **ReCoN**, short for *Replication of Complex Networks* and described in Section 3, uses and extends ideas of LFR, a generator used for benchmarking community detection algorithms. Using the original degrees and a found community structure we are able to capture a much-more detailed signature of the network than a parametrization of the LFR generator. In Section 4 we discuss the generative models that we use for comparison (among them RMAT, HUDG, and BTER) and develop model fitting schemes for them.

Our comparative experimental study in Section 5 indicates that ReCoN performs overall quite well and usually better than other generators in terms of realism. We can also conclude that the ReCoN implementation is fast, as it is capable of creating realistic scaled replicas on the scale of 10^8 edges in minutes. The ReCoN code is publicly available in the open-source network analysis package **NetworkKit** [31].

2 Realistic Replicas

We consider a generative model realistic if there is high structural similarity between the synthetic graphs produced and relevant real-world networks. It is neither our goal nor generally desirable to obtain an exact correspondence between original and replica. First, this would exclude the use case of obfuscation. Secondly, obtaining an isomorphic graph is rarely required for generalizable experiments. Note that we consider a single “realism score” for each model inappropriately reductionist. Rather, we quantify diverse aspects of realism in our experimental evaluation and leave it to the reader to decide about their relative importance.

For 1-scale replicas (with the same size as the original), we measure the similarity in terms of a set of commonly used metrics: Sparsity (number of edges vs number of nodes); degree distribution (more precisely its Gini coefficient); maximum degree as a proxy for the connectedness of hub nodes; average local clustering coefficient to measure the local presence of triangles; diameter to monitor the small-world effect; number of connected components and number of communities as additional non-local features. These metrics cover both local and global properties and are deemed important characteristics of networks [23].

How can we extend the notion above regarding realism to *scaled* replicas of a network? To answer this question, let us look at the scaling behavior of a set of 100 Facebook social networks [32]. These networks were collected at an early stage of the Facebook online social networking service in which networks were still separated by universities. Fig. 1 plots basic structural measures of these Facebook networks against the number of nodes n , as well as a regression line and confidence intervals (shaded area) to emphasize the trend. While linear regression may not always seem completely appropriate for these data, the general trend is still captured.

We can observe from Fig. 1 a growth of the number of edges m that is linear in n , an increase in the skew of the node degree distribution as measured by the Gini coefficient, a growing maximum node degree, a slightly falling average local clustering coefficient, a nearly constant small diameter of the largest connected component, and a slightly growing number of connected components (which can be explained

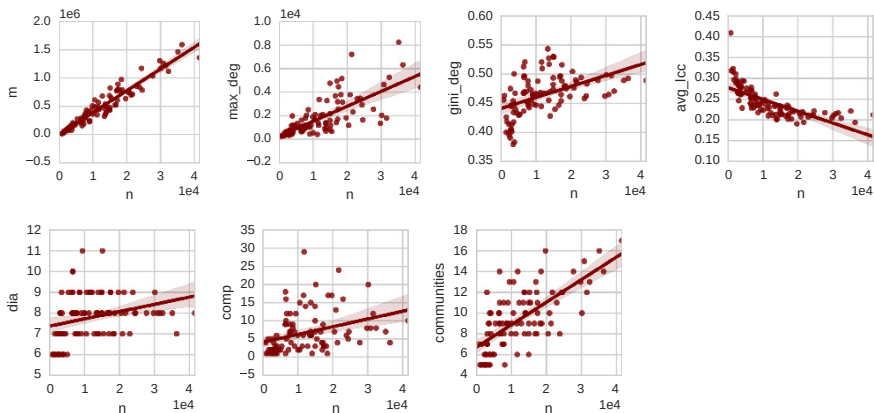


Fig. 1: Scaling behavior of 100 Facebook networks; from left to right and top to bottom: number of edges, maximum degree, Gini coefficient of degree distribution, average local clustering coefficient, diameter, number of components, number of communities found by PLM

by some small connected components that exist in addition to a giant component). We detect communities using PLM (Parallel Louvain Method), a modularity-based community detection heuristic [30], and report the number of communities minus the number of these small connected components. It can be observed that the number of non-trivial communities grows slightly.

While we do not propose that these scaling laws are universal, the trends represented here are commonly observed [4, 5, 27]. Thus, we use them to define desired scaling properties for the remainder of the study as follows: m grows linearly with n ; the diameter does not change significantly, preserving the “small world property”; the shape of the degree distribution remains skewed; the maximum node degree increases; the number of connected components may grow; the number of communities increases slightly.

Recall that one use case for our generator is testing of graph and network analysis algorithms. Since the running time is an essential feature in such tests, we also consider a realistic replication of running times important. To this end, we select a set of graph algorithms that (i) compute important features of networks and are thus frequently used in network analysis tasks and that (ii) cover a variety of patterns of computation and data access, each of which may interact differently with the graph structure. The set consists of algorithms for connected components (essentially breadth-first search), PageRank (via power iteration), betweenness approximation (according to Geisberger et al. [11]), community detection (PLM, [30]), core decomposition (according to [9]), triangle counting (according to [15]), and spanning forest (essentially Kruskal’s algorithm without edge weights).

3 The Generation Algorithm ReCoN

We introduce ReCoN, a generator for replicating and scaling complex networks. Its input is a graph and a community structure on it. For fitting a given graph without given community structure, we use PLM [30] in order to detect a community structure first. The basic idea of ReCoN is to randomize the edges inside communities and the edges between communities while keeping the node degrees. This happens separately such that each community keeps as many edges as it had before. For scaling a graph, we first create as many disjoint copies of the graph as desired and then apply the aforementioned steps. During the randomization of the edges between the communities the copies usually become connected with each other.

The idea of randomizing graphs inside and between communities is inspired by the LFR generator, a benchmark graph generator for community detection algorithms [19]. There the basic building blocks are also a random subgraph per community and a global graph. However, in the LFR generator the degrees and communities are not given but generated using a power law degree distribution and a power law community size distribution with nodes assigned to communities at random, while ReCoN uses the given graph as input for them.

For randomizing graphs while preserving the degree sequence we use random edge switches where two edges $\{u, v\}$, $\{y, z\}$ chosen uniformly at random are changed into $\{u, z\}$, $\{y, v\}$ if the resulting graph is still simple, i. e. does not contain any duplicate edges or self-loops. Similar to the edge switching implementation provided by [33] we use 10 times the number of edges as the number of random edge switches. Previously performed experiments (e. g. [22]) have shown that this is enough to expect the resulting graph to be drawn uniformly at random from all graphs with the given degree sequence.

For an original graph $O = (V, E)$ with $n_o = |V|$ nodes and a desired scaling factor x , ReCoN executes the following steps:

1. Detect a community structure $\mathcal{C} = \{C_1, \dots, C_k\}$ on O using PLM.
2. Create H as the disjoint union of x copies of O . The community structure is also copied such that the new community structure $\mathcal{D} = \{D_1, \dots, D_{x \cdot k}\}$ consists of $x \cdot k$ communities, i. e. each copy of O gets its own copy of the community structure that is aligned with the structure of the copied graph.
3. For each community D_i , $1 \leq i \leq x \cdot k$, randomize the edges of the subgraph $H[D_i]$ that is induced by the community D_i while keeping the degree distribution using random edge switches.
4. Randomize the remaining edges, i. e. all edges in H that are not part of one of the subgraphs $H[D_i]$ using random edge switches. Note that afterwards some edges that were not in one of the $H[D_i]$ can now be inside a community. In order to avoid this, rewiring steps are performed by executing edge switches of such forbidden edges with random partners. A similar step is also used in the LFR generator where it was observed that in practice only few rewiring steps are necessary [18].

Note that it is not necessary to start with the original graph in step 3 and 4. Using any graph with the same degree sequence is enough as the result is random

anyway. Therefore, it is enough to know a community structure (as opposed to the whole original graph) and for each node the internal and external degree, i. e. how many neighbors it has inside and outside its community, respectively. For our implementation we choose this alternative. Further, we execute step 3 in parallel for all communities as the subgraphs are disjoint.

In addition to replicating important properties with high fidelity, the randomization in step 3 and 4 naturally produces random variance among the set of replicas.

4 Fitting Generative Models to Input Graphs

Parametrized generative models require fitting schemes for learning parameters from the original network. Because, usually, such schemes are not unique, exploring them would be important future work. For this study, we have chosen one scheme per model, parameters of which are summarized in Table 1 in the full version of this paper [28]. Below we discuss a fitting scheme for power law degree distributions, and briefly describe the generative models that are compared with ReCoN.

Fitting power law degree distribution (PLD). We apply our custom power law fitting scheme. A practical replication of a network requires preserving the original average (otherwise, the density will be changed) as well as minimum and maximum degrees (applications can be sensitive to such fundamental properties as degree-1 nodes and the distribution of hubs). In general, it is assumed (and implemented in many algorithms [8]) that PLD only holds starting with a minimum degree and that for smaller degrees, the distribution might be different. As the LFR generator only generates a plain PLD, we cannot apply this assumption. Therefore, we fit the PLD exponent such that, with the given minimum and maximum degree, the average degree of the real network is expected when a degree sequence is sampled from this PLD. Using binary search in the range of $[-6, -1]$, we repeatedly calculate the expected average degree until the power law exponent is accurate up to an error of 10^{-3} .

Erds–Rnyi, Barabasi-Albert, Chung-Lu and ESMC. *Erds–Rnyi* random graphs (ER) [24] are fundamental and an important baseline with the edge probability parameter that we set to produce the same edge-to-node ratio as in O . The *Barabasi–Albert* model (BA) [2] implements a preferential attachment process by which a PLD emerges, which has been claimed to be a typical feature of real complex networks. In BA, we set the number of edges coming with each new node to fit the original edge-to-node ratio. The *Chung-Lu* (CL) model [1] recreates a given degree sequence in expectation. The *Edge-Switching Markov Chain Generator* (ESMC) generates a graph that is randomly drawn from all graphs with exactly the given degree sequence (see e.g. [22], [26]). In both CL and ESMC we use the original degree sequence. To generate larger networks, x copies of this sequence are concatenated, multiplying the number of nodes by x while keeping the relative frequency of each degree.

RMAT. The *Recursive Matrix* (RMAT) model [7] was proposed to recreate various properties of complex networks, including an optional power-law degree distribution, the small-world property and self-similarity. The RMAT model can only generate

graphs with 2^s nodes, where s is an integer scaling parameter. In order to target a fixed number of nodes n_r , we calculate s so that $2^s > n_r$ and delete $2^s - n_r$ random nodes. The choice of other parameters as well as the running time of fitting are discussed in [28].

Hyperbolic Unit Disk Graphs (HUDG). The random hyperbolic graph model embeds nodes into hyperbolic geometry and connects close nodes with higher probability [17]. The unit-disk variant HUDG we use in this paper connects only nodes whose distance is below a certain threshold. We are focussing on the unit-disk variant to be able to use a very fast generator for this model [21]. The model has been shown to replicate some properties observed in real networks, such as a power-law degree distribution. This method receives as parameters the desired number of nodes, the average degree of the original network and a power law exponent which is fitted as described above. As the given power law exponent must be larger than 2, we supply at least an exponent of 2.1.

BTER. This method receives a degree distribution and the desired clustering coefficient per degree, i.e., for each degree to be realized the number of occurrences and the average clustering coefficient per degree. For scaled replicas we scale the occurrences of all degrees by the scaling factor. This leads to the target number of nodes while also preserving the general shape of the degree distribution. In order to retain the distribution of the clustering coefficients, we leave them unchanged while scaling the network.

LFR. LFR was designed as a benchmark graph generator for community detection algorithms [19]. Apart from the number of nodes it requires parameters for power law distributions of the node degrees and the community sizes, and a mixing parameter that determines the ratio between intra- and inter-cluster edges. We detect communities using PLM [30] and fit the parameters for the two power law distributions as described above using the original degree sequence and the found community sizes. The mixing parameter is set to the ratio between intra- and inter-cluster edges of the found communities. The details are described in [28].

5 Computational Experiments

Our implementations of ReCoN and the various fitting methods are based on NetworkKit [31], a tool suite for scalable network analysis. It also contains many of the generators we use for comparison and provides a large set of graph algorithms we use for our experiments. NetworkKit combines C++ kernels with an interactive Python shell to achieve both high performance and interactivity, a concept we use for our implementations as well. All implementations are freely available as part of the package at <https://networkkit.iti.kit.edu>. This also includes a faster and parallel implementation of the LFR generator (compared to the original implementation [10]).

Our experimental platform is a shared-memory server with 256 GB RAM and 2x8 Intel(R) Xeon(R) E5-2680 cores at 2.7 GHz, using the GCC 4.8 compiler and the openSUSE 13.1 OS. More technical details are available in [28].

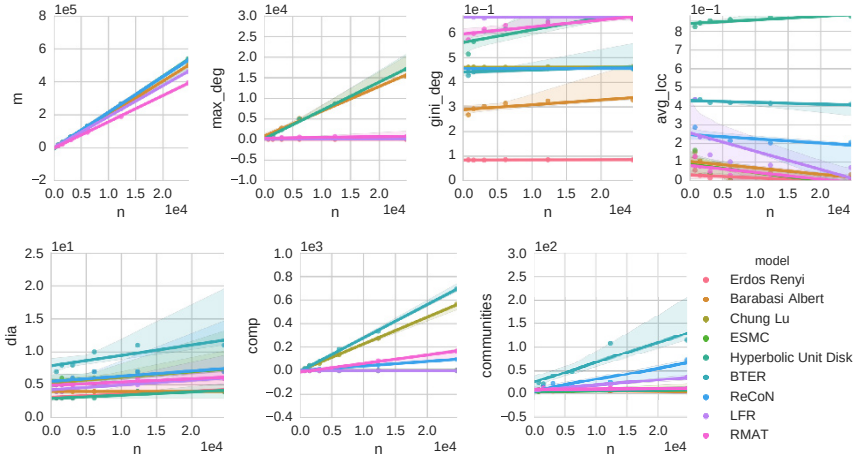


Fig. 2: Scaling behavior of the different generators on the fb-Caltech36 network. From left to right and top to bottom: number of edges, max. degree, Gini coefficient of the degree distribution, average local clustering coefficient, diameter, number of components, number of communities.

As described in Section 2, we are interested in how well the different generators replicate certain structural features of the original networks as well as the running times of various graph algorithms. The results are described subsequently.

Scaling behavior of the generators. The following experiments consider the scaling behavior of generative models. Given the parametrization discussed before, we look at the evolution of structural features with growing scale factor x up to $x = 32$. We consider the same basic scalar features as for the real networks in Sec. 2 and, due to space constraints, point to [29] for more results.

In Figure 2, we show the results of the scaling experiments for the fb-Caltech36 network. The number of edges of the replicas is increased almost linearly by all generators to $\approx 5 \cdot 10^5$ edges which approximately corresponds to 32 times the edges of the original network. Therefore, all generators seem to keep the average degree of the original network, which is expected as it is a parameter of all considered generators. Surprisingly, the maximum degree strongly increases up to 10 or 15 thousand with HUDG and BA generators, respectively. The original maximum degree is 248, so that the new value is even significantly higher than the scaled maximum degree (i. e. $248 \cdot 32$). Actually, from the scaling study in Sec. 2, we could expect an increase, but rather in a lower range, so the degree distribution of BA and HUDG generators are not realistic. Concerning the Gini coefficient, one can clearly see that ER does not generate a skewed degree distribution at all. All generators that get the exact degree sequence as input keep the Gini coefficient constant, which is expected and also relatively realistic from our scaling study.

The original average local clustering coefficient of 0.43 is almost exactly reproduced by BTER in which it is an input parameter. The HUDG method increases it

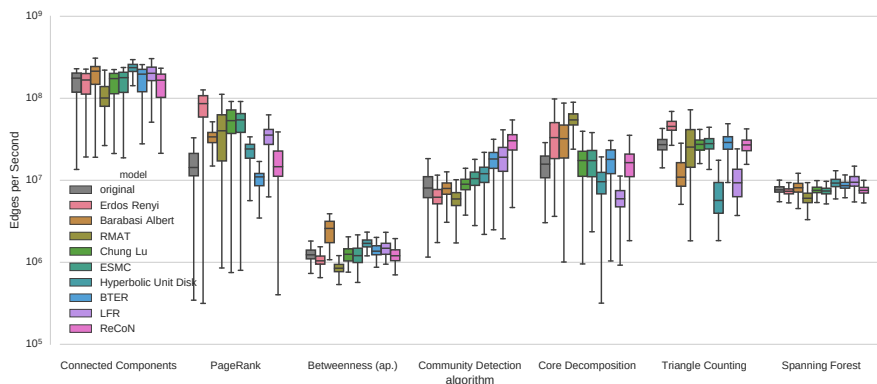
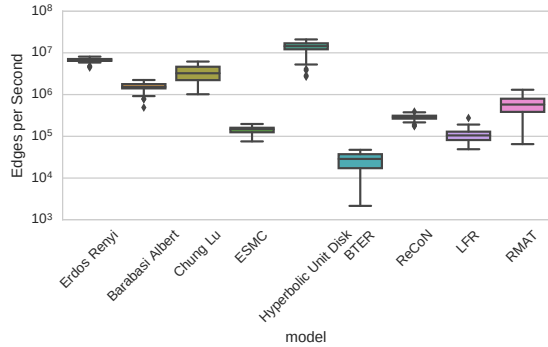


Fig. 3: Running time replication of a set of network analysis algorithms. Running times are in edges per second, i.e., higher is faster.

to 0.8, most others obtain very small values. Our new ReCoN generator is less far off with 0.25 and a slightly decreasing clustering coefficient; the latter is actually realistic as we saw in Sec. 2. LFR is able to generate a clustering coefficient above 0.2 initially. Other generators produce much lower clustering coefficients. The original diameter of 6 is almost exactly kept by ReCoN, all other generators except BTER produce networks with slightly lower diameters, while BTER generates networks whose diameter is almost twice bigger. All generators show a slight increase of the diameter when the networks are larger, which is consistent with our scaling study. While most generators produce networks with just a single connected component, CL and BTER generate a large number, RMAT and ReCoN a moderate number of connected components. In the case of CL, BTER and RMAT, this is probably due to a large number of degree-0 nodes. The original network consists of a giant component and 3 small components; ReCoN scales them linearly, which is due to its parametrization. The original network is split into eight non-trivial communities, that number should increase slowly according to Sec. 2. Only in the networks generated by BTER, ReCoN and LFR, PLM can find a significant and increasing amount of communities. While PLM finds over 100 non-trivial communities in the network generated by BTER, there are fewer communities detectable in the networks generated by ReCoN and even less in the ones generated by LFR. Overall, ReCoN is the only generator that keeps the degree distribution, and produces a realistic clustering coefficient and a small diameter while keeping the graph connected and preserving a moderate number of communities. All other generators are either unable to keep the diameter or the connectivity or the number of communities. It is part of future work to investigate whether the full hyperbolic random graph model can alleviate the weaknesses of the unit-disk case.

Replicating running times of graph algorithms. Synthetic graphs are frequently used in algorithm engineering to estimate the running time of an algorithm assuming that this time will be similar on real networks. We examine if this is indeed the case with the generative models we consider. Using the previously described generators

Fig. 4 Fitting and generating: processing speed measured in edges/s (size of replica graph / total running time, measured on 100 Facebook graphs)



and fitting schemes, we generate replicas of 100 Facebook networks and test a variety of graph algorithms (see Sec. 2) on both the original and replica sets.

Our experiments demonstrate (see Fig. 3) that the running times on the replica sets often do not match those on the original set. The gray segments of the box plots represent the distribution of running times measured on a set of original networks. Ideally, the distribution on the synthetic networks would be identical. The difference is statistically nontrivial, though. Small variance between the models exists for connected components and spanning forest computations, since their running time is nearly constant per edge. Other algorithms exemplify how much running time can depend on network structure, especially community detection, core decomposition, triangle counting and PageRank. In general, the running time measurements obtained on ReCoN match the originals closely in most cases. An exception is community detection, where PLM seems to profit from ReCoN’s explicit model of communities. BTER shows close matches, too.

Generator running times. In Fig. 4, we show the running times of parameter fitting and generating a replica for all methods. Processing speed is given in the number of edges per second. The entire set of Facebook networks was used to produce the measurements, so generated replicas range from about 15000 to 1.5 million edges. For all models, generating the graph takes up the vast majority of time. BTER’s MATLAB-based implementation is slowest, while the ER and HUDG generators are the fastest. Our implementations of LFR and ReCoN are not among the fastest generators, but fast enough to produce millions of edges in minutes.

6 Conclusion

We have presented a new generator, ReCoN, for replicating and scaling existing networks. In an extensive experimental evaluation (not all results could be shown due to space constraints, see [28, 29] for more results) we have shown that ReCoN is capable of generating networks which are (i) similar to the original network in terms of important structural measures and (ii) lead to similar running times of many graph and network analysis algorithms. Using ReCoN it is possible to realistically replicate an existing network, and to scale the synthetic version by orders of magnitude, e. g., in

order to test algorithms on larger data sets where they are not available. Furthermore, it allows to create anonymized copies of such networks that can be distributed freely and allow to conduct representative experiments on them. While other generators sometimes perform better concerning certain criteria, none of the other generators is capable of approximately reproducing such a wide range of properties and running times.

Acknowledgements This work is partially supported by the DFG under grants ME 3619/3-1 and WA 654/22-1 within the Priority Programme 1736 *Algorithms for Big Data*, and by the NSF awards #1522751 and #1647361. Funding was also provided by *Karlsruhe House of Young Scientists* via the *International Collaboration Package*.

References

- [1] Aiello, W., Chung, F., Lu, L.: A random graph model for massive graphs. In: Proceedings of the thirty-second annual ACM symposium on Theory of computing, pp. 171–180. Acm (2000)
- [2] Albert, R., Barabási, A.: Statistical mechanics of complex networks. *Reviews of modern physics* **74**(1), 47 (2002)
- [3] An, W.: Fitting ERGMs on big networks. *Social Science Research* **59**, 107 – 119 (2016). Special issue on Big Data in the Social Sciences
- [4] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: Structure and dynamics. *Physics reports* **424**(4), 175–308 (2006)
- [5] Caldarelli, G., Vespignani, A.: Large scale structure and dynamics of complex networks. World Scientific (2007)
- [6] Chakrabarti, D., Zhan, Y., Faloutsos, C.: R-MAT: A recursive model for graph mining. In: Proc. 4th SIAM Intl. Conf. on Data Mining (SDM). SIAM, Orlando, FL (2004)
- [7] Chakrabarti, D., Zhan, Y., Faloutsos, C.: R-MAT: A recursive model for graph mining. Computer Science Department p. 541 (2004)
- [8] Clauset, A., Shalizi, C.R., Newman, M.E.: Power-law distributions in empirical data. *SIAM review* **51**(4), 661–703 (2009)
- [9] Dasari, N.S., Ranjan, D., Zubair, M.: ParK: An efficient algorithm for k-core decomposition on multicore processors. In: 2014 IEEE International Conference on Big Data, Big Data 2014, Washington, DC, USA, October 27-30, 2014, pp. 9–16. IEEE (2014)
- [10] Fortunato, S.: Benchmark graphs to test community detection algorithms. URL <https://sites.google.com/site/santofortunato/inthepress2>
- [11] Geisberger, R., Sanders, P., Schultes, D.: Better approximation of betweenness centrality. In: ALENEX, pp. 90–100. SIAM (2008)
- [12] Goldenberg, A., Zheng, A.X., Fienberg, S.E., Airoldi, E.M.: A survey of statistical network models. *Foundations and Trends® in Machine Learning* **2**(2), 129–233 (2010)
- [13] Gutfraind, A., Meyers, L., Safro, I.: Musketeer: Multiscale entropic network generator. URL <https://people.cs.clemson.edu/~isafro/musketeer/index.html>
- [14] Gutfraind, A., Safro, I., Meyers, L.A.: Multiscale network generation. In: 18th International Conference on Information Fusion, FUSION 2015, Washington, DC, USA, July 6-9, 2015, pp. 158–165 (2015)
- [15] Hamann, M., Lindner, G., Meyerhenke, H., Staudt, C.L., Wagner, D.: Structure-preserving sparsification methods for social networks. *Social Netw. Analys. Mining* **6**(1), 22:1–22:22 (2016)
- [16] Kolda, T.G., Pinar, A., Plantenga, T., Seshadhri, C.: A scalable generative graph model with community structure. arXiv preprint arXiv:1302.6636 (2013)

- [17] Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A., Boguñá, M.: Hyperbolic geometry of complex networks. *Physical Review E* **82**, 036,106 (2010)
- [18] Lancichinetti, A., Fortunato, S.: Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E* **80**(1), 016,118 (2009)
- [19] Lancichinetti, A., Fortunato, S., Radicchi, F.: Benchmark graphs for testing community detection algorithms. *Physical Review E* **78**(4), 046,110 (2008)
- [20] Leskovec, J., Faloutsos, C.: Scalable modeling of real graphs using kronecker multiplication. In: Proc. 24th Intl. Conference on Machine learning, pp. 497–504. ACM (2007)
- [21] von Looz, M., Meyerhenke, H., Prutkin, R.: Generating random hyperbolic graphs in sub-quadratic time. In: Algorithms and Computation - 26th International Symposium, ISAAC 2015, Nagoya, Japan, December 9-11, 2015, Proceedings, pp. 467–478 (2015)
- [22] Milo, R., Kashtan, N., Itzkovitz, S., Newman, M.E.J., Alon, U.: On the uniform generation of random graphs with prescribed degree sequences. eprint arXiv:cond-mat/0312028 (2003)
- [23] Newman, M.: Networks: an introduction. Oxford University Press (2010)
- [24] P. Erdős, A.R.: On the Evolution of Random Graphs. Publication of the Mathematical Institute of the Hungarian Academy of Sciences (1960)
- [25] Robins, G., Pattison, P., Kalish, Y., Lusher, D.: An introduction to exponential random graph (p^*) models for social networks. *Social Networks* **29**(2), 173 – 191 (2007). Special Section: Advances in Exponential Random Graph (p^*) Models
- [26] Schlauch, W.E., Horvát, E.Á., Zweig, K.A.: Different flavors of randomness: comparing random graph models with fixed degree sequences. *Social Network Analysis and Mining* **5**(1), 1–14 (2015). DOI 10.1007/s13278-015-0267-z
- [27] Snijders, T.A.: The statistical evaluation of social network dynamics. *Sociological methodology* **31**(1), 361–395 (2001)
- [28] Staudt, C., Hamann, M., Safro, I., Gutfraind, A., Meyerhenke, H.: Generating Scaled Replicas of Real-World Complex Networks. Tech. rep., arXiv (2016). URL <http://arxiv.org/abs/1609.02121>. ArXiv:1609.02121
- [29] Staudt, C.L.: Algorithms and software for the analysis of large complex networks. Ph.D. thesis, Karlsruhe Institute of Technology (2016). DOI 10.5445/IR/1000056470
- [30] Staudt, C.L., Meyerhenke, H.: Engineering parallel algorithms for community detection in massive networks. *IEEE Trans. on Parallel and Distributed Systems* **27**(1), 171–184 (2016)
- [31] Staudt, C.L., Sazonovs, A., Meyerhenke, H.: NetworKit: A tool suite for large-scale network analysis. *Network Science To appear*
- [32] Traud, A.L., Mucha, P.J., Porter, M.A.: Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications* **391**(16), 4165–4180 (2012)
- [33] Viger, F., Latapy, M.: Random generation of large connected simple graphs with prescribed degree distribution. In: 11th International Conference on Computing and Combinatorics. Kunming, Yunnan, Chine (2005)

Modeling of Data Communication Networks using Dynamic Complex Networks and its Performance Studies

Suchi Kumari and Anurag Singh

Abstract To study the underlying organizing principles of various complex systems, designing an efficient graph-based model for data representation, is a fundamental aspect. As the topological structure of the network changes over time, it is a challenging task to design a communication system having ability to respond to randomly changing traffic. We are interested to find out the suitable and fair traffic flow rates to each system for getting optimal system utility using dynamic complex network model. In this context, we design and simulate a growth model of the data communication network based on the dynamics of in-flowing links which is motivated by the concept that newly added node will connect to the most influential nodes already present in the system. The connectivity distribution of the evolved communication networks follows power law form, free from network scale. We analyze Kelly's optimization framework for a rate allocation problem in communication networks at different time instants, and optimal rates are obtained with the consideration of arbitrary communication delays.

Key words: Complex Networks, Dynamic Networks model, Communication Processes, System Utility

1 Introduction

Systems such as social, telecommunication, computer, biological, citation, etc. can be modeled as a graph considering distinct elements represented by nodes and there is a connection (links) between them. The graph has nontrivial topological properties, connections between elements are neither purely regular nor purely random. These systems are very large, can be modeled in the form of a network,

Suchi Kumari (e-mail: suchisingh@nitdelhi.ac.in) · Anurag Singh (e-mail: anuragsg@nitdelhi.ac.in)✉

Department of Computer Science and Engineering, National Institute of Technology, Delhi, Delhi-110040, India

helps us to understand the behavior of the system, called complex networks. Complex networks are currently being studied across many fields of science systems in nature. In complex networks [2, 11, 15, 16], links often exhibit various features: they can be directed, have different weights assigned to it, be active only at certain times. The demographic features of random graphs using the probabilistic approach in network structure analysis was developed by Erdos and Renyi (ER), they investigated random network model [6].

Watts and Strogatz (WS) have proposed a model, which generates complex network having small world properties [22]

The more complex network model, Scale-free model was proposed by Barabasi-Albert ([1]). The model is defined in two steps:

- Expansion: Starting with a small number (n_0) of nodes, at each instant of time a new node appears with $a(\leq n_0)$ links which are connected to the existing nodes in the system.
- Preferential connection: The Π probability that a newly added node will be attached to node i only when the value of influential parameter (k_i) of that node is maximal.

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}$$

After time t , the network will contain total $n = t + n_0$ nodes and at links. Network evolves into a scale invariant case and hence the scaling exponent is independent of a total number of links a .

Limitations of BA model are as follows:

- Both invariant, expansion and preferential connections are compulsory.
- It is assumed that new connection is established only when new nodes are added to the system. But, in real life, connections are made continually.
- In some systems, re-association or rewiring of the existing links can happen, and they are also following preferential connection, but if reattachment dominates over expansion, then this will destroy the behavior, i.e., the power-law scaling in the system.

To make the network dynamic, an important ingredient of the dynamics is a preferential connection of links (outflowing/inflowing). Tadic [20] has focused on outflowing links and shown that both the outflowing and inflowing links follow a heavy-tailed distribution with distinct exponents. Momentary alteration of the outflowing links inside the networks effect on both the outflowing and inflowing links. After establishing a correlation between the outflowing and inflowing links, it is shown that the local structure of the network is qualitatively different compared to the case without an update. The expansion, as well as update, are taking place at unique time scale, a new node $n(= t)$ appears in the network (expansion), and a number $X(t)$ of new links are scattered. There is an increasing interest in investigating not only the process dynamics on networks [18, 19] but also the dynamics of networks [7]. There is a need to extend the basic network concept to include time relations between

nodes arose, leading to many models for Time-Varying Graphs (TVGs) [5, 10, 21]. Although the nodes are placed in the space randomly, network structure depends on the distribution of links.

The structure of connections has an immediate impact on the accessibility of particular node, and it is the backbone for the stability of the network. If the number of connected components increases, then there must be at least one path between each pair of the node. Social networks are one of the examples of dynamic network where, people are represented by nodes and if two people are connected then, there will be a connection between them. Contacts are not static, it is temporal and depends on the state(active/inactive) of nodes. Some activity parameter is used to generate temporal links and an adaptive network is formed by incorporating memory effect to know about past connections. In [3], reciprocal action of individual activity and network structure are shown. State of the node determines the dynamic activity of human interaction and states are also decided by the connection between nodes.

Another example is communication networks, which can respond to randomly changing traffic flow rates by reassigning traffic routes and by reallocating resources. As expansion and updates, both are happening at unique time scale, so the design and control of such kind of network is a challenging task. Topology is changing at each time-stamp. Due to change in topology, the performance of the network is also affected [12]. The exponent is independent of a total number of links a .

Modern communication networks are faced with multiple challenges at different layers and modeling their rate control behavior [9, 13, 14, 17] with volatile and dynamic connectivity setting is a prominent issue. Real life network settings are extremely volatile, and still communication takes place albeit with degraded quality and possible setback in performance. There is a new kind of thinking to understand the underlying reasons for volatile spatiotemporal behavior and how one can re-engineer them for optimal performance for this change.

Rather than closing our eyes to these kinds of hard technical difficulties, a framework is proposed to model arbitrarily changing directed networks in both space and time with the help of proposed mathematical models in [1, 20]. It is shown that the degree distribution of the networks follows the power law and hence scale free in nature. We analyze Kelly's optimization framework for a rate allocation problem in communication networks at different time instants, and optimal rates are obtained considering user's willing to pay and network cost.

Section 2 states about mathematical modeling of the network, Section 3 provides a real life mobile communication network examples with arbitrary link changes by maintaining certain set of rules and followed by algorithmic steps, Section 4 presents a numerical example illustrating the algorithm and Section 5 describes the conclusion and explains the future directions of this work.

2 Mathematical model and related work

In this section we give a brief description about rate allocation problem. We contemplate a network with a set E of links and a set of R users. Let C_e be the capacity

of the link, where, $e \in E$. For each user $k \in R$, a route r_k has been assigned for a particular time instant $t_i \in T$, where $t_i \mid 1 \leq i \leq \tau$ contains a nonempty subset of E . A zero-one matrix A of the size $E \times R \times t$ is defined where, $A_{k,e,t} = 1$, if e is in the route of user k at time t , otherwise zero. When the user k is assigned a rate $x_{k,t}$ then utility of user k at rate $x_{k,t}$ is given as $U_{k,t}(x_{k,t})$ is increasing, strictly concave function of $x_{k,t}$ over the range $x_{k,t} \geq 0$. Aggregate utility is calculated by summing up all utilities of user k at rate $x_{k,t}$ and is denoted as $\sum_{k \in R, t \in T} U_{k,t}(x_{k,t})$. Rate allocation problem can be formulated as the following optimization problem.

$$\begin{aligned} & \text{SYSTEM}(U_t, A_t, C_t) \\ & \text{maximize } \sum_{k \in R, t \in T} U_{k,t}(x_{k,t}) \\ & A_n^T x_t \leq C_t \text{ and } x_t \geq 0 \end{aligned} \quad (1)$$

where, $n = (1, 2, \dots, \tau)$, τ is the total number of time instants. A_n is the matrix formed in the time interval t_{n-1} to t_n . The constraint shown above tells us that the flow through a link can not exceed the capacity of particular link [8]. For handling large scale of the system, it is inconvenient to allocate each user an optimal rate. Hence, Kelly has divided this problem into two simpler problems named as user's optimal problem and network's optimal problem [9]. Let each user k is demanded a price per unit flow as λ_k . A user chooses an amount to pay at per unit time is $P_k(t)$ according to the incurred cost with the user. Hence, user receives a flow, $x_k(t) = P_k(t)/\lambda_k$ then user's optimal price will be

$$\begin{aligned} & \text{User}_k(U_k(t), \lambda_k(t)), \\ & \text{maximize } U_k(x_k(t)) - p_k(t), \\ & p_k > 0 \end{aligned} \quad (2)$$

On the other hand, network wants to maximize weighted log function of $p_k(t)$. Therefore, network utility function can be written as

$$\begin{aligned} & \text{NETWORK}(A_t, C_t, p_t), \\ & \text{maximize } \sum_{k \in R, t \in T} P_k(t) \log(x_k(t)), \\ & A_n^T x_t \leq C_t \text{ and } x_t \geq 0. \end{aligned} \quad (3)$$

The values of λ_k, P_k and x_k are considered variable with time. Each user in the network, $k \in R$ initially computes the price per unit flow by using the Eqn. (4) and it is willing to pay, $P_k(t)$. It adjusts its rate based on the feedback provided by the links in the network. Each user attempts to make equilibrium by its willingness to pay the total price for the complete duration. Finally, one can always find out unique stable value of the price per unit flow λ_k^* , rate x_k^* and willingness to pay and P_k^* and corresponding convergence vectors will be $\lambda^* = \lambda_k^*, k \in R, P^* = P_k^*, k \in R$ and $x^* = x_k^*, k \in R$.

For each user, k is given price per unit flow as λ_k and the amount for which user is willing to pay, $P_k(t)$ at time t . Hence, the rate assigned to user k is $x_k(t) = P_k(t)/\lambda_k$. Utility of each user k at a particular time instant is assumed by strictly concave function of users rate at that time instant. Suppose that each user adopts a rate based

flow control. At each time instant each link $e \in E$ charges a price per unit flow of $\mu_e(t) = g_e(\sum_{k:e \in E} x_k(t))$ where $g_e(\bullet)$ is an increasing function of the total flow through it and $g_e(y)$ is

$$g_e(y) = c_e \cdot (y/C_e)^\omega$$

where, c_e is constant and assumed one, C_e is the capacity of resource $e \in E$. The defined price function arises when resources are modeled as $M/M/1$ queue. $M/M/1$ queue is a queue having some length with the single server. Processes are arriving with certain rate and then service is provided to that process by the server. Suppose processes are arriving at rate λ and μ is the service rate. Hence, $\rho = \lambda/\mu$, where ρ is the average proportion of time when the server is occupied or busy. C_e is the service rate and packet will receive a mark when there is already ω packets in the queue. Now consider the following system of differential equation

$$\frac{dx_k(t)}{dt} = \sigma_k(P_k(t) - x_k(t) \sum_{e \in E} \mu_e(t)) \quad (4)$$

Each user firstly computes it's willingness to pay as $P_k(t)$ then it adjusts its rate based on the feedback provided by the links in the network and trying to balance its willing to pay and total price. Eqn. (4) consists of two components: a steady increase in the rate proportional to $P_k(t)$ and steady decrease in the rate proportional to the feedback provided by the network.

3 Proposed work

Like the Internet, communication networks use a specific set of rules to connect the components and directed links are used to access data. In the communication network, degree distribution of both out-flowing and in-flowing links follow a heavy tail distribution with separate exponent values. In the proposed model, we have given preference for in-flowing link because the newly created link is attached to the node which has highest in-flowing link probability. Set of rules which are used in the formation of dynamic networks, yield that the distributions of both out-flowing and in-flowing links are interdependent. Another important feature of the model is that the connection between pairs of nodes is not fixed in time, but it may change on the time scale of the network's expansion(updates of links).

Here, a communication network is formed with scale-free property by modifying the BA model [4] and model [20]. The modified directed network is formed by maintaining the following rules.

1. Directed nature of linking.
2. Expansion and update are done at unique time scale. At each time unit t , a new node $n(=t)$ is added to the network (expansion) and total number $X(t)$ of new connections are established and allocated to the nodes. Newly created links are divided into two groups: added link and updated link. Distribution of the links is done using following rules specified below.

- Enter the value of fraction β , γ , such that $\beta < 1$ and $0.5 < \gamma \leq 1$.
- A fraction $f_\beta(t) = \beta X(t)$ of new links are out-flowing links from the new appeared node $n = t$ and added with the nodes existing in the network at $(t - 1)$ based on priority, here β is a fraction with $\beta < 1$.
- Another remaining fraction $f_{(1-\beta)}(t) = (1 - \beta)X(t)$ are the updated (removed and rewired) links within existing nodes excluding the newly added nodes.

Updated links may have two types:

- A fraction $f_{up}(t) = \gamma f_1(t)$ links are rewired with the value of fraction γ , $0.5 < \gamma \leq 1$. It helps to maintain the growing nature of the network.
- Fraction $f_{dl}(t) = (1 - \gamma)f_1(t)$ are removed from the network.
- The parameter δ is the ratio of updated and added links in the model and is given by $\delta = \frac{f_1(t)}{f_0(t)} = \frac{1-\beta}{\beta}$, which is independent of the added number of links $X(t)$ and known as **correlation parameter**.

3. We can define two functions preferential update and preferential attachment.

While talking about communication network, the concept of preferential linking driven by the demand of the node for the flowing data into the network. In addition to this, preference for the update is given to only a few nodes, rather than updating all nodes at each time instant. Moreover, some of the nodes want to update out-flowing links more frequently than others. Apart from the newly appeared node, larger update probability is given to most active nodes at time t , i.e., an out-flowing link from the node $k \leq n$ appears according to preferential attachment. Removal of links are done randomly but the rearrangement of links done based on preferential attachment.

Algorithmic steps are given for expansion and updation of network.

Attributes of links contain *linkid*, named, delay and capacity. We have to send packets from multiple sources to multiple destinations based on shortest path. Shortest path is measured in terms of hop count. Multiple users can send data from specific source (S) to destination (D) based on shortest path and these S-D sets are generated according to user's choice. If number of users increases, then the congestion level will increase according to the selection of paths.

Initially, shortest path for user is found and after that optimal data rate of the user is calculated by using these steps:

Algorithm 1 Network Evolution

```

1: Input: A small number ( $m_0$ ) for seed network ,  $m(\leq$ 
    $m_0)$  for distribution,  $\beta, \gamma$  and timer.
2: Output: Evaluated network.
3: while  $T \leq \textit{timer}$  do
4:   Add a node at each time instant.
5:   for  $m: 1$  to  $f_\beta(t)$  do
6:     Select a node of higher probability to attach with.
7:   end for
8:   for  $n: 1$  to  $f_{up}(t)$  do
9:     Select an arbitrary source and link it to the node having higher inflowing
       link probability.
10:  end for
11:  for  $p: 1$  to  $f_{dt}(t)$  do
12:    Randomly select  $v$  a link to remove.
13:  end for
14: end while

```

Algorithm 2 Finding shortest path and optimal rate for each user

```

1: for  $i := 1$  to  $\textit{numPair}$  do
2:   Find shortest path between source and destination
3:   for  $j := 1$  to  $\textit{numofNode}$  do
4:     Calculate frequency of occurring of active node during path formation
5:      $\textit{rate}(j) = \frac{\textit{capacity}}{\textit{frequency}(j)}$ ;
6:   end for
7: end for
8: for  $r = 1$  to  $\textit{numofPair}$  do
9:   Update feedback for each element of S-D pair
10:   $\textit{ratePath}(r) = \textit{minRate}(\textit{elementofPath})$ ;
11:   $A(r) = \textit{rand}(1, 10)$ ;
12:   $\textit{Wpay}(r) = \textit{ratePath}(r) * (\frac{a}{\textit{ratePath}(r)+b})$ ;
13:   $\textit{meu1}(r) = \textit{meu}$ ;
14: end for
15: Use the value of  $\textit{ratePath}$ ,  $A$ ,  $\textit{Wpay}$  and  $\textit{Meu1}$  to find out the rate of convergence of each user.

```

Evolution of the network is done at a unique time instant. Here we have taken initial size of the network of $(100 + m)$ nodes i.e., $t_0 = 100 + m$ units and $\delta t = 100$, hence $t_{i+1} = t_i + \delta t$ and the series will look like $T = (t_0, t_1, t_2, \dots, t_\tau)$ and the value is, $T = (100 + m, 200 + m, 300 + m, \dots, 100\tau + m)$. Each user firstly computes and shows a willingness to pay as $P_k(t)$ then it adjusts its rate based on the feedback provided by the links in the network and trying to balance it is willing to pay the total price. Eqn. (4) consists of two components: a steady increase in the rate proportional to $P_k(t)$ and steady decrease in the rate proportional to the feedback provided by the network. Initial values of willingness to pay for the user, feedback of the network and the rate of the resources are provided to the solver for finding out the optimal

rate of each user. At each time instant user increases its willingness to pay but due to congestion in the network rate and becomes stable after some time.

4 Simulation and results

In most of the real world networks, the degree of the majority of nodes has low value, but there exist few hub nodes, having a high degree. Some social networks are found to have degree distributions that approximately follow a heavy-tailed distribution: $P(k) \sim k^{-\alpha}$, where $2 < \alpha \leq 3$, known as scale-free networks. In a scale-free network, numerous nodes with few links coexist with a few hub nodes, having connected with thousands or even millions of links. To make all the values for large k visible use of a log-log plot is needed. We can either use logarithmic axes, with powers of 10 or we can plot $\log p_k$ in function of $\log k$. Here, logarithmic axes, with powers of 10 is taken for plotting the probability distribution of node degrees over the whole network and the degree distribution shows power law behavior. The value of β can be obtained from δ as $\beta = \frac{1}{(1+\delta)}$. There are four possible cases of the value of the δ , depending on updated and newly added link in the network.

In Fig. 1, it is shown that evolved network follows power law degree distribution when network has different values of nodes along with correlation parameter δ .

1. $\delta = 0 (\beta = 1)$ i.e, only expansion is happening no update (rearrangement and removal). The degree distribution of the network having $N = 10000$ nodes and scaling exponent $\alpha = 2.664$, is shown in Fig. 1(d).
2. $\delta < 1 (\beta > 0.5)$, more number of new links are getting added than updated. The degree distribution of the networks having $N = 10000$ nodes and the values of $\beta = 0.6$ (expansion), $\gamma = 0.5$ (rearrangement) and $\alpha = 2.455$, shown in Fig. 1 (b).
3. $\delta > 1 (\beta < 0.5)$, more number of links are updated than added. Degree distribution of the networks having $N = 10000$, $\beta = 0.25$, $\gamma = 0.7$ and $\alpha = 2.065$ is shown in Fig. 1(c).
4. $\delta = 1$, when both the value of updated and added links are same, degree distribution of the networks with $N = 10000$, $\beta = 0.5$, $\gamma = 0.5$ and $\alpha = 2.486$ is shown in Fig. 1(a).

From the graph shown in Fig. 1, it is analyzed that, by increasing the parameter β in the range $(0, 1)$, corresponds to decrease of the correlation parameter δ in the interval $(\infty, 0)$, the slope of the distributions increases.

The network is formed using the algorithm 1. Evolved network is formed by putting the values of parameters as: size of the seed network $m_0 = 5$, Number of links which is distributed at each time instant $m (\leq m_0)$, β , γ and *timer*. User's routes for sending packets are varying according to time. At each time instant, a new node appears with m links and expansion as well as re-arrangements are done. As the network becomes larger and larger, many paths are available for sending packets for each user between desired source and destination. All routes are equally weighted hence, users can select any of these routes for sending packets.

Each user can send data along one of the shortest paths to the destination with a

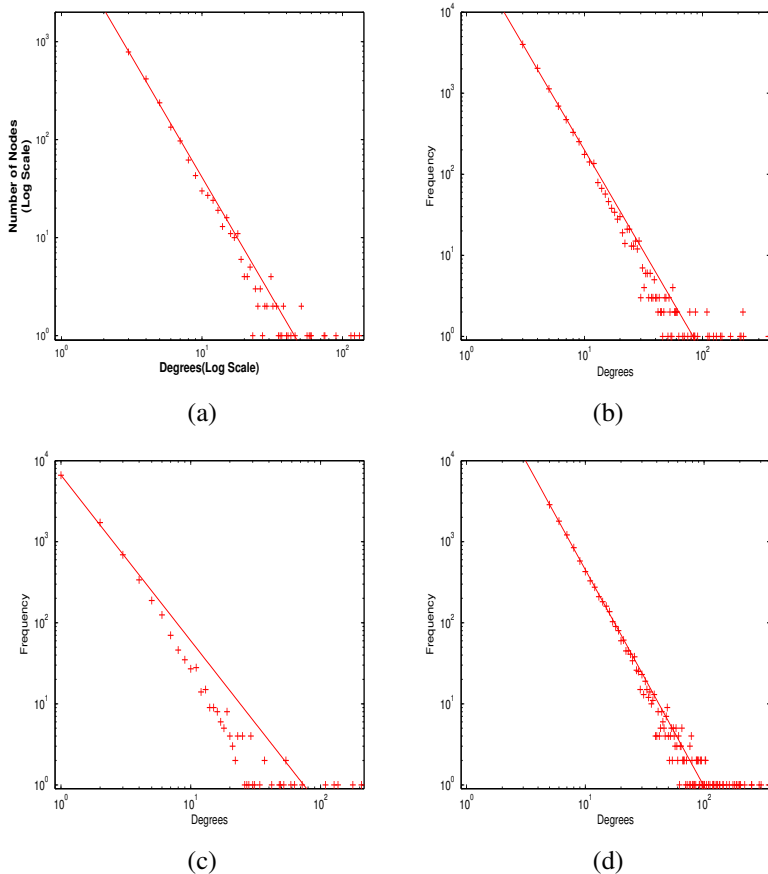


Fig. 1: Degree distribution of the network when number of nodes are and average ratio of updated and newly added links are **(a)** $N = 10000$, $\delta = 1$, **(b)** $N = 10000$, $\delta = 0.67$, **(c)** $N = 10000$, $\delta = 3$ and **(d)** $N = 10000$, $\delta = 0$

maximum flow rate of individual links. Multiple users need to share the resources hence, data sending rate got reduced, and it can no more send data with a maximum rate. User’s rate depends on two parameters; it’s own willingness to pay and network’s feedback. Using rate control theorem given in (4), an optimal data sending rate of each user is obtained. In Fig. 2, User1’s and User2’s data sending rates are shown at different time instants. Instead of, increased network size, optimal rates are not increasing. User rates depend on the demand of particular resources coming in the shortest route. If demand is high, then data sending rate will be less.

Multiple users want to establish connections between a distinct pair of nodes and hence, a shortest possible communication path is chosen. There may exist a multiple number of shortest routes having the same number of hop count, but betweenness

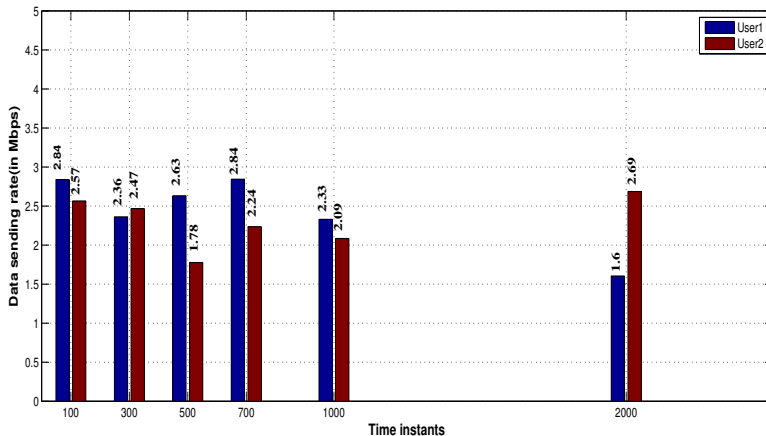


Fig. 2: Conservation of data sending rates of User1 and User2 at different time instants

centrality of all shortest paths would not be same. Hence, data flow rate of the paths having high betweenness value will be less. Optimal rates are also dependent on betweenness. User’s optimal rates along with their betweenness values are shown in Table 1. User’s optimal rates are also shown in figure 3.

Table 1: User’s optimal rate through the shortest routes having different betweenness values(maximum and minimum), when number of nodes $N = 100$

	Source	Destination	Betweenness (Minimum)	Betweenness (Maximum)	Optimal Rates (Minimum Betweenness)	Optimal Rates (Maximum Betweenness)
User1	7	38	0.1931	0.1966	6.316335	7.737758
User2	77	96	0.0906	0.5877	3.383048	4.446214
User3	21	37	0.0616	0.3066	6.489473	9.356067
User4	68	79	0.0856	0.3872	4.409398	5.202014
User5	13	36	0.0751	0.4118	7.057041	8.913657
User6	20	47	0.0185	0.1467	5.134327	6.963006
User7	36	62	0.0608	0.1006	5.519844	6.575628
User8	24	65	0.085	0.3955	3.780126	7.232818
User9	40	6	0.0762	0.2017	6.978428	12.434885
User10	18	75	0.1483	0.3413	5.832557	7.118301
User11	24	85	0.0818	0.3847	4.081002	8.247708
User12	39	2	0.2144	0.3314	9.000121	11.311826

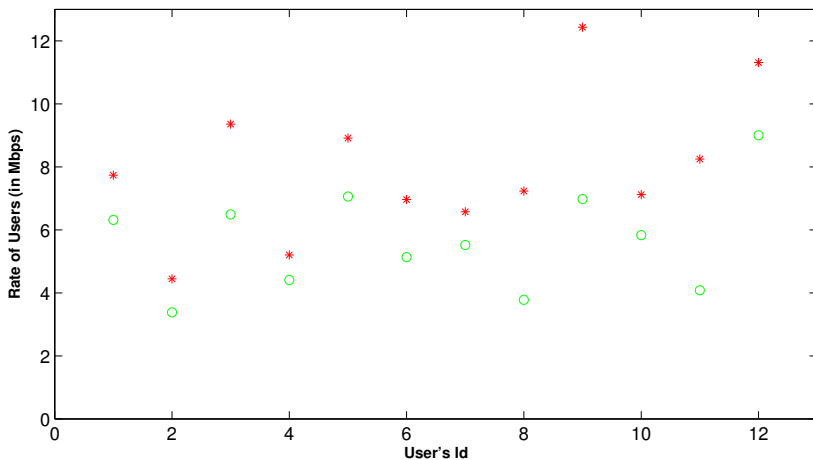


Fig. 3: User's optimal rate through the shortest routes having different betweenness values(maximum and minimum), when number of nodes $N = 100$

5 Conclusions and Future directions

In this paper, a model is proposed to represent complex dynamic systems in the form of complex networks and their representation is also given by using mathematical expression. The proposed model is simple, flexible and efficient for the representation and modeling of dynamically changing networks. At each time instance, a new node appears with few links, either for expansion or update based on the value of fractions β and γ . Expansion and update (removal and rewiring) of links are done based on the preferential basis (most influential nodes). Network changes at each time instant and it grows according to the value of time. Various experiments are performed for finding out the topological structure of the evolved network and the rate control behavior is also studied. At each time slot, user's route changes and hence data sending rates also change accordingly. Rate control theorem proposed by Kelly [9], formulated for static network, is used for obtaining optimal user data sending rates to maximize the system utility.

In this paper User's willingness to pay is taken as constant value and it is proportional to the initial capacity(maximum) of that User. It can vary dynamically according to the rate assigned to the User. We have not considered the role of delays while solving System utility. User's routes are selected by considering shortedness, betweenness centrality and initial capacity of users are taken according to their in-degree. It can be extended by considering different objective functions by using parameters such as reputation, influence etc.

References

- [1] Adamic, L.A., Huberman, B.A.: Power-law distribution of the world wide web. *science* **287**(5461), 2115–2115 (2000)
- [2] Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. *Reviews of modern physics* **74**(1), 47 (2002)

- [3] Aoki, T., Rocha, L.E., Gross, T.: Temporal and structural heterogeneities emerging in adaptive temporal networks. *Physical Review E* **93**(4), 040,301 (2016)
- [4] Barabási, A.L., Albert, R., Jeong, H.: Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications* **272**(1), 173–187 (1999)
- [5] Casteigts, A., Flocchini, P., Quattrociocchi, W., Santoro, N.: Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems* **27**(5), 387–408 (2012)
- [6] Erdős, P., Rényi, A.: On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci* **5**(17-61), 43 (1960)
- [7] Golubitsky, M., Stewart, I.: Nonlinear dynamics of networks: the groupoid formalism. *Bulletin of the american mathematical society* **43**(3), 305–364 (2006)
- [8] Kelly, F., Voice, T.: Stability of end-to-end algorithms for joint routing and rate control. *ACM SIGCOMM Computer Communication Review* **35**(2), 5–12 (2005)
- [9] Kelly, F.P.: Mathematical modelling of the internet. *Mathematics unlimited-2001 and beyond* pp. 685–702 (2001)
- [10] Kim, H., Anderson, R.: Temporal node centrality in complex networks. *Physical Review E* **85**(2), 026,107 (2012)
- [11] Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. *Journal of Complex Networks* **2**(3), 203–271 (2014)
- [12] Kumari, S., Singh, A., Ranjan, P.: Towards a framework for rate control on dynamic communication networks. In: *Proceedings of the International Conference on Internet of Things and Cloud Computing, ICC '16*, pp. 12:1–12:7. ACM, New York, NY, USA (2016). DOI 10.1145/2896387.2896397. URL <http://doi.acm.org/10.1145/2896387.2896397>
- [13] La, R.J., Anantharam, V.: Utility-based rate control in the internet for elastic traffic. *IEEE/ACM Transactions on Networking (TON)* **10**(2), 272–286 (2002)
- [14] La, R.J., Ranjan, P.: Asymptotic stability of a primal algorithm with a finite communication delay. In: *Decision and Control, 2006 45th IEEE Conference on*, pp. 644–649. IEEE (2006)
- [15] Newman, M.: Complex systems: A survey. *arXiv preprint arXiv:1112.1440* (2011)
- [16] Newman, M.E.: The structure and function of complex networks. *SIAM review* **45**(2), 167–256 (2003)
- [17] Ranjan, P., La, R.J., Abed, E.H.: Global stability conditions for rate control with arbitrary communication delays. *IEEE/ACM Transactions on Networking (TON)* **14**(1), 94–107 (2006)
- [18] Singh, A., Singh, Y.N.: Nonlinear spread of rumor and inoculation strategies in the nodes with degree dependent tie strength in complex networks. *Acta Physica Polonica B* **44**(1), 5–28 (2013)
- [19] Singh, A., Singh, Y.N.: Rumor dynamics with inoculations for correlated scale free networks. In: *Communications (NCC), 2013 National Conference on*, pp. 1–5. IEEE (2013)
- [20] Tadić, B.: Dynamics of directed graphs: the world-wide web. *Physica A: Statistical Mechanics and its Applications* **293**(1), 273–284 (2001)
- [21] Tang, J., Scellato, S., Musolesi, M., Mascolo, C., Latora, V.: Small-world behavior in time-varying graphs. *arXiv preprint arXiv:0909.1712* (2009)
- [22] Watts, D.J., Strogatz, S.H.: Collective dynamics of small-world networks. *nature* **393**(6684), 440–442 (1998)

Testing for the signature of policy in online communities

Alberto Cottica, Guy Melançon and Benjamin Renoust

Abstract Most successful online communities employ professionals, sometimes called “community managers”, for a variety of tasks including onboarding new participants, mediating conflict, and policing unwanted behaviour. We interpret the activity of community managers as network design: they take action oriented at shaping the network of interactions in a way conducive to their community’s goals. It follows that, if such action is successful, we should be able to detect its signature in the network itself. Growing networks where links are allocated by a preferential attachment mechanism are known to converge to networks displaying a power law degree distribution. Our main hypothesis is that managed online communities would deviate from the power law form; such deviation constitutes the signature of successful community management. Our secondary hypothesis is that said deviation happens in a predictable way, once community management practices are accounted for. We investigate the issue using empirical data on three small online communities and a computer model that simulates a widely used community management activity called *onboarding*. We find that the model produces in-degree distributions that systematically deviate from power law behavior for low-values of the in-degree; we then explore the implications and possible applications of the finding.

Alberto Cottica (e-mail: alberto@cottica.net)
University of Alicante, Alicante, Spain & Edgeryders, Brussels, Belgium

Guy Melançon (e-mail: Guy.Melancon@u-bordeaux.fr)
University of Bordeaux, LaBRI CNRS UMR 5800, Bordeaux, France

Benjamin Renoust (e-mail: renoust@nii.ac.jp)
National Institute of Informatics & JFLI CNRS UMI 3527, Tokyo, Japan

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no 688670.



1 Introduction

Organizations running online communities typically employ community managers, tasked with encouraging participation and resolving conflict [18]. Only a small number of the participants (one or two members in the smaller communities) will recognize some central command, and carry out its directives. We shall henceforth call such directives *policies*. Putting in place policies for online communities is costly, in terms of recruitment, training, and software tools. This raises the question of what benefits organizations running online communities expect from policies; and why they choose certain policies, and not others.

Online communities can be modeled as social networks of interactions across participants, and organizations can be modeled as economic agents maximizing some objective function (*e.g.* profit, welfare). Hence the topology of the interaction network affects the ability for participants to contribute to the maximization of the target variable. For example, Facebook is constantly rewiring the interaction network across its users to ensure better targeted and more effective advertising, therefore enhancing their revenue [21].

Such organizations choose their policies such as community managers could take action to change the network towards maximizing their objective function.

All this implies that the decision to deploy a particular policy on an online community is a network design exercise. An organisation decides to employ a community manager to shape the interaction network of its community in a way that helps its own ultimate goals. And yet, interaction networks in online communities cannot really be designed; they are the result of many independent decisions, made by individuals who do not respond to the organization's command structure. An online community management policy is then best understood as an attempt to "influence" emergent social dynamics; to use a more synthetic expression, it can be best understood as the attempt to design for emergence. Its paradoxical nature is at the heart of its appeal.

We are interested in detecting the mathematical signature of specific policies in the network topology. We consider a simple policy called *onboarding* [18, 19]. As a new participant becomes active (*e.g.* by posting her first post), community managers are instructed to leave her a comment that contains (a) positive feedback and (b) suggestions to engage with other participants that she might share interests with.

We model online conversations as social networks, and look for the effect of onboarding on the topology of those networks. We proceed as follows:

1. We initially examine data from three small online communities. Only two of them deploy a policy of *onboarding*. We observe that, indeed, the shape of the degree distribution of these two differs from that of the third.
2. We propose an experiment protocol to determine whether onboarding policies can explain the differences observed between the degree distributions of the first two online communities and that of the third one.
3. Based on the generalized model [10] we simulate the growth of online communities. Variants to the model cover the relevant cases: the absence of onboarding policies and their presence, with varying degrees of effectiveness.

4. We run the experiment protocol against the degree distributions generated by the computer model, and discuss its results.

Section 2 briefly examines the two strands of literature that we mostly draw upon. Section 3 presents some data from real-world online communities; it then proceeds to describe our main experiment, a computer simulation of interaction in online communities with and without onboarding. Section 4 presents the experiment's results. Section 5 discusses them.

2 Related work

Collective intelligence [15] scholars confirmed importance of online community management practices, indeed, they have tried to systematize it [9] and produce technological innovation to support it [8, 20]. These tools are meant to facilitate and encourage participation to online communities, to make it easier for individuals to extract knowledge from them. Studying human communities is a traditional focus of network science [5, 6], for which easily available datasets of online communities make an ideal ground for structural analysis: friendship in Facebook [16, 17], following/retweet/mentions for Twitter [11, 12, 13], or vote and comments in discussions [11, 14, 22, 23].

Starting in the 2000s, online communities became the object of another line of enquiry, stemming from network science. Network representation of relationships across groups of humans has yielded considerable insights in social sciences since the work of the sociometrists in the 1930s, and continues to do so; phenomena like effective spread of information, innovation adoption, and brokerage have all been addressed in a network perspective [5, 6]. As new datasets encoding human interaction became available, many online communities came to be represented as social networks. This was the case for social networking sites, like Facebook [16, 17]; microblogging platform like Twitter [11, 12, 13]; news-sharing services like Digg [11]; collaborative editing projects like Wikipedia [14]; discussion forums like the Java forum [23]; and bug reporting services for software developers like Bugzilla [22]. Generally, such networks represent participants as nodes. Edges represent a relationship or interaction. The nature of interaction varies across online communities: one edge can stand for friendship for Facebook; follower-followed relationship, retweet or mention in Twitter; vote or comment in Digg and the Java forum; talk in Wikipedia; comment in Bugzilla.

In contrast to collective intelligence scholars, network scientists typically do not address the issue of community management, and treat social networks drawn from online interaction as fully emergent. In this paper, we employ a network approach to investigate the issue of whether the work of community managers leaves a footprint detectable by quantitative analysis. To our knowledge, no other work attempted this investigation. In particular, we exploit a result from the theory of evolving networks, from seminal work by Barabási and Albert [2] showing that the assumption of growth and preferential attachment, when taken together, result in a network whose degree distribution converges to a power law ([1, 3]). The model was later generalized in

various ways and tested across a broad range of networks, including social networks [10].

We use this generalization as a baseline state. The degree distribution of the interaction network in an online community follows a power law by default. The action of online community managers, as they attempt to further the goals of the organisation that runs the online community, will result in its degree distribution deviating from the baseline power law in predictable ways. Such deviation can be interpreted as the signature that the policy is working well.

The most important difficulty with this method is the absence of a counterfactual: if a policy is enacted in the online community, the baseline degree distribution corresponding to the absence of the policy is not observable, and viceversa. This rules out a direct proof that the policy “works”. Hence our choice to combine empirical data and computer simulations.

3 Materials and methods

In this section we introduce the empirical data, the experiment protocol and the simulation model we use in the experiment.

3.1 Empirical data

We examine data from three real-world online communities: InnovatoriPA is a community of (mostly) Italian civil servants discussing how to introduce and foster innovation in the public sector. It does not employ any special onboarding or moderation policy. Edgeryders is a community of (mostly) European citizens, discussing public policy issues from the perspective of grassroots activism and social innovation. It adopts the onboarding of new members policy. Matera 2019 is a community of (mostly) citizens of the Italian city of Matera and the surrounding region, discussing the city’s policies. It also adopts the onboarding policy.

The communities are modeled as interaction networks (summarized in Table 1) in which nodes are users and edges represent directed comments from A to B , weighted by the number of comments written. A glance at their respective visualizations (Figure 1) suggests that the networks of the three communities have very different topologies. Innovatori PA displays more obviously visible hubs than the other two.

We fitted power laws in-degree distributions of these three online communities, as of early December 2014. Next, we tested the hypothesis that degree distributions follow a power law, as predicted by [10]. To do so, we first fitted power functions to the entire support of each in-degree distribution¹. We next fitted power functions to the right tail of each in-degree distribution, *i.e.* for any degree $k(n) \geq k_{min}$, where

¹ We emphasize in-degree, as opposed to out-degree, because directedness is implicit in the idea of preferential attachment, and because the in-degree distribution is the one to follow a power law in online conversation networks ([10]).

	Innovatori PA	Edgeryders	Matera2019
Policy	"no special policy"	"onboard new users"	"onboard new users"
In existence since	December 2008	October 2011	March 2013
Accounts created	10,815	2,419	512
Active participants (nodes)	619	596	198
Number of edges (weighted)	1,241	4,073	883
Average distance	3.77	2.34	2.51
Maximum degree	155	238	46
Average degree	2.033	6.798	4.454
Goodness-of-fit for $k \geq 1$			
exponent	1.611	1.477	1.606
p -value	0.21	0.00 (reject)	0.00 (reject)
Goodness-of-fit for $k \geq k_{min}$			
k_{min}	2	5	6
exponent	1.834	2.250	2.817
p -value	0.76	0.45	0.94

Table 1: Comparing interaction networks of the three online communities and testing for goodness-of-fit of power functions to degree distributions. "Exponent" refers to the power law's scaling parameter. "p-value" to the result of the test that the degree distribution of the community was generated by a power law with that exponent.

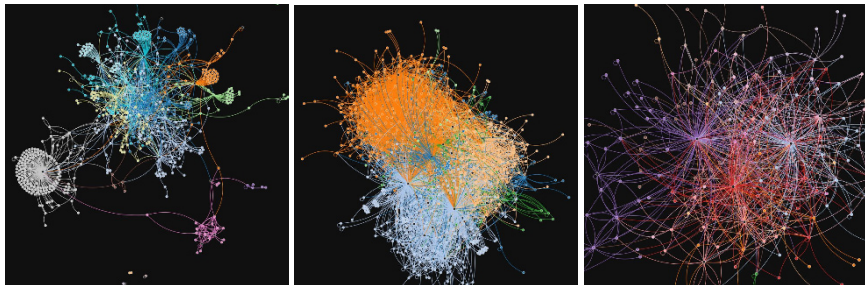


Fig. 1: Interaction networks of three small online communities. Innovatori PA (left) does not have an onboarding policy in place, whereas the two others do (Edgeryders: center, Matera: right).

k_{min} is the in-degree that minimizes the Kolmogorov-Smirnov distance (hereafter denoted as D) between the fitted function and the data with in-degree $k \geq k_{min}$.

Finally, we ran goodness-of-fit (hereafter *GoF*) tests for each in-degree distribution and for fitted power functions. The method we followed throughout the paper is borrowed from Clauset *et al* [7]. The null hypothesis tested is that the observed distribution is generated by a power function with exponent α . We compare the D statistic of the observed distribution with those of a large number of synthetic datasets drawn by the fitted power function. Such comparison is summarized in a p -value, that indicates the probability of the D statistic to exceed the observed value conditional to the null hypothesis being true. p -values close to 1 indicate that the power function is a good fit for the data: the null hypothesis is not rejected. p -values close to zero

indicate that the power function is a bad fit for the data, and reject the null hypothesis. The rejection value is set, conservatively, at 0.1. Results are summarized in Table 1.

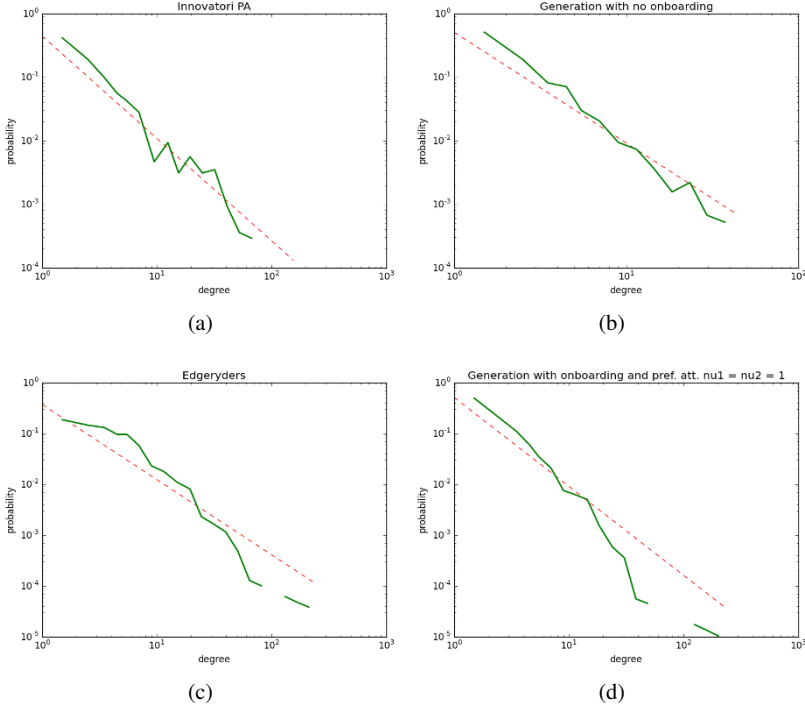


Fig. 2: (log - log) Probability density function from the degree distributions of: (a) the Innovatori PA network without onboarding policy in place versus (b) a simulated network with preferential attachment and no onboarding. (c) The Edgeryders network with onboarding and preferential attachment versus (d) a simulated network with preferential attachment and fully effective onboarding ($\nu_1 = \nu_2 = 1$).

As we consider the interval $k \geq 1$, we find that the in-degree distribution of the Innovatori PA network – the unmoderated one – is consistent with the expected behavior of an evolving network with preferential attachment. We cannot reject the null hypothesis that it was generated by a power law. For other two communities, both with onboarding policies, the null hypothesis is strongly rejected. On the other hand, when we consider only the tail of the degree distributions, i.e. $k \geq k_{min}$, all three communities display a behavior that is consistent of a setting with preferential attachment.

These results are consistent with the objectives of the onboarding policy, consisting in helping newcomers find their way around a community that they don't know yet. A successfully onboarded new user will generally have some extra interaction with existing active members. All things being equal, we can expect extra edges to appear in the network, and interfere with the in-degree distribution that would appear in the

absence of onboarding – explaining the non-power law distribution of Edgeryders and Matera2019. Extra edges target mostly low connectivity nodes: onboarding targets newcomers, and focuses on helping them through the first few successful interactions. Highly active (therefore highly connected) members do not need to be onboarded. This may explain why all three communities display power law behavior in the upper tail of their in-degree distributions, regardless of onboarding.

3.2 Experiment protocol

The difference observed between the two communities with onboarding policies and the one without might be caused not by the policy itself, but by some other unobserved variable. To explore the policy’s effects, we generate and compare computer simulations of interaction networks in online communities that are identical except for the presence and effectiveness of onboarding policies.

Communities are assumed to grow over time, with new participants joining them in sequence. At each point in time, new edges appear; their probability of targeting an existing node grows linearly with that node’s in-degree. Additionally, communities might have or not have onboarding policies. See section 3.3 below for a specification of onboarding in the model.

We generated 100 communities with no onboarding policy (control group), 100 communities for each couple of v_1 and v_2 in $\{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ (treatment group), and computed their in-degree distribution. Next, we tested two hypotheses for the 3700 networks generated.

- *Hypothesis 1.* The in-degree distribution of C is generated by P for any $k \geq 1$.
- *Hypothesis 2.* The in-degree distribution of C is generated by P for any $k \geq k_{min}$.

Where C is the synthetic network; $k(s)$ is the in-degree of a node s ; k_{min} is the in-degree that minimizes the Kolmogorov-Smirnov distance D between the fitted function and the data over $k \geq k_{min}$; and P is the best-fit power-law model for the in-degree distribution of C . We expect non-rejection of both hypotheses for the control group; and rejection of Hypothesis 1, but not of Hypothesis 2, in case of effective onboarding (high v_1) in the treatment group.

3.3 Simulation

We simulated the growth of network in an online community with and without onboarding following preferential attachment [2] in the generalized model [10].

Without onboarding: A network is initialized with two reciprocally connected nodes. At each step a new node (new user) is introduced, and m new edges (comments) are also created, with a uniformly random picked source. The probability that the new edge points to a node s is proportional to $k(s) + A_s$ where $k(s)$ is the in-degree of node s and A_s is a parameter representing additional attractiveness of the node.

With onboarding: Network initialization and growth are as in the case of no onboarding. Additionally, an edge targeting the newly created node is added at each step.

This edge represents the action of the community manager, addressing a welcome message to the newcomer. At this point of each step, with probability $v_1 \in [0, 1]$, a new edge is added with source as the new node (the newcomer becomes active). The edge’s target is chosen by preferential attachment, as described previously². Next (still in the same step), with probability $v_2 \in [0, 1]$, another edge is added with a uniformly picked source and the newcomer node as target. This represent the online community acknowledging the newcomer by addressing her a comment.

We call v_1 *onboarding effectiveness*. It is the probability of the newcomer to react to the community manager’s onboarding activity. We call v_2 *community responsiveness*. It is the probability for the new participant to have attracted the attention of other participants and engage in a conversation. We set network size to 2000 nodes; $m = 1$; and $A_s = 1$ for all nodes, in the tradition of [2] and [10].

4 Results

4.1 Goodness-of-fit of the power-law model

For each network evolved we computed two best-fit power-law models, one for $k \geq 1$ and the other for $k \geq k_{min}$ where k_{min} is the in-degree the minimizes D between the fitted function and the data over $k \geq k_{min}$. On each of these models, we ran a *GoF* test as in section 3.1, results are reported in Table 2.

We first examine the case in which $k \geq 1$. We conclude that onboarding seems to have some effect on the goodness-of-fit of the generated data to their respective best-fit power-law models. When onboarding is introduced, fewer degree distributions, out of our 100 runs, are power law-shaped; also, the average p-values returned by *GoF* tests are lower than those of the control group. Running *t*-tests of the null hypothesis that the average *p*-value in the control group is equal to the average *p*-values in the treatment group results in a strong rejection for any combination of v_1 and v_2 .

We now turn to the question of the role played by v_1 and v_2 within the treatment group. Figure 3 (a, b) shows the cumulated density functions of the p-values in the control and treatment groups as v_1 and v_2 vary. Increasing onboarding effectiveness v_1 pushes average p-values of the *GoF* tests down, making it less likely that Hypothesis 1 would be rejected. Increasing community responsiveness v_2 seems not to play any role at all. This is somewhat surprising. Recall that we modeled onboarding as the command-and-control creation of an extra edge at each step, targeting newcomers to the online community. This has a strong negative effect on the p-value returned by the *GoF* test (compare any p-value in Table 2 with the p-value of the control group with no onboarding). When a responsive community adds a second edge, however, there is no additional effect on the p-value. This result is confirmed by regression analysis (not shown here).

² The source of the new edge is irrelevant to the model’s results, since we only study in-degree. We specify it in the text to help exposition, since the expected result of onboarding is the activation of newcomers.

Table 2: Average p -values (number of rejections) for *GoF* tests of power-law models to in-degree distributions of interaction networks in online communities. Control group communities have no onboarding (control group). Power-law models are estimated over all nodes with degree $k \geq 1$

	Control group: 0.262688 (23)					
	$v_2 = 0.0$	$v_2 = 0.2$	$v_2 = 0.4$	$v_2 = 0.6$	$v_2 = 0.8$	$v_2 = 1$
$v_1 = 0.0$	0.0593 (83)	0.0601 (81)	0.0520 (83)	0.0479 (88)	0.0551 (82)	0.0514 (85)
$v_1 = 0.2$	0.0629 (78)	0.0797 (73)	0.0852 (70)	0.0834 (73)	0.0834 (73)	0.0796 (70)
$v_1 = 0.4$	0.1047 (66)	0.0970 (65)	0.0986 (61)	0.0831 (69)	0.0829 (76)	0.1157 (56)
$v_1 = 0.6$	0.0964 (59)	0.0855 (67)	0.1021 (63)	0.1269 (51)	0.0906 (70)	0.0797 (71)
$v_1 = 0.8$	0.1326 (55)	0.1152 (60)	0.1036 (66)	0.1091 (61)	0.1188 (60)	0.1228 (61)
$v_1 = 1$	0.1009 (65)	0.1207 (62)	0.1326 (54)	0.1164 (60)	0.1230 (54)	0.1205 (57)

When $k \geq k_{min}$, the effect of introducing onboarding on the *GoF* disappears. Over 99% of the networks in the treatment group give rise to distributions that turn out to be a good fit for a power-law model when k_{min} is chosen so as to minimize D between the degree distributions themselves and their best-fit power-law models. We conclude that Hypothesis 2 cannot be rejected, regardless of whether onboarding is present or not.

4.2 Lower bounds

We find a limited, albeit statistically significant, effect of onboarding on the value of k_{min} , the value of k that minimizes D between the data generated by the computer simulation and the best-fit power-law model. Figure 3(c,d) shows that over 60% of the in-degree distributions from interaction networks in the control group, vis-a-vis only 30 to 40% of those in the treatment group, fit a power-law model best for $k_{min} \leq 3$. Within the treatment group, some variability is associated to the increase of v_1 , whereas v_2 does not seem to play a significant role. Regression analysis (not shown here) shows that, once we control for the presence of onboarding, neither parameter is significant.

4.3 Exponents

Introducing onboarding to an online community has a positive and significant effect on the value of the exponent of the best-fit power-law model for the in-degree distribution of its interaction network. This is consistent with previous studies ([10]). This result holds when the best-fit power-law models is computed over $k \geq k_{min}$, where k_{min} is the value of k that minimizes D between the simulated in-degree distribution and its best-fit power-law model. When it is computed over the whole support of the in-degree distribution ($k \geq 1$), it also holds, except for $v_1 = 1$. Table 3 illustrate the average value of the scaling parameter α , and the p -value of a t -test on

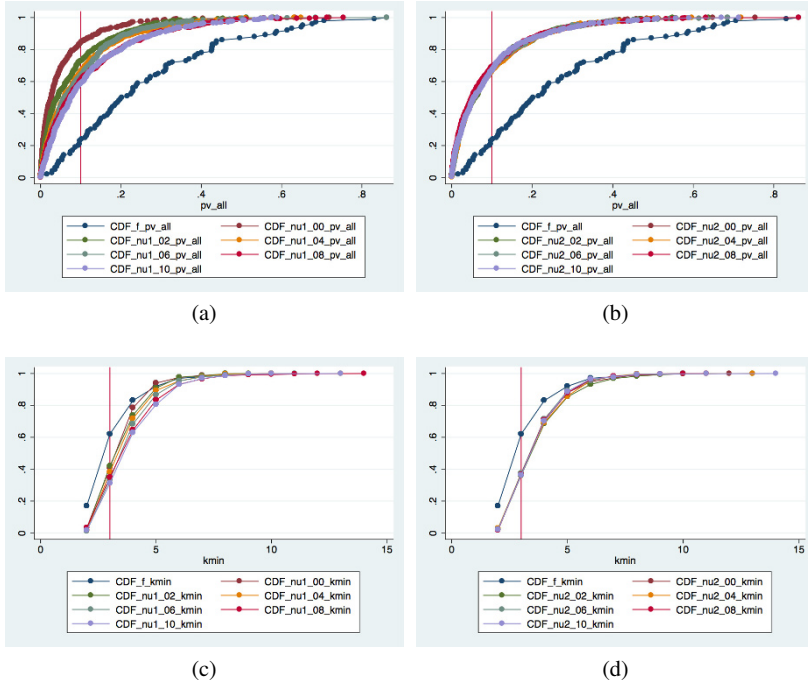


Fig. 3: (a,b): CDF of p-values returned by *GoF* tests to the (best-fit) power-law models for in-degree distributions of the interaction networks in the control and treatment groups. 20% of the networks evolved without onboarding (dark blue) have degree distributions that test negatively for H_1 . When onboarding is introduced, it rises to between 50 and 90%. (a,c) the treatment group interaction networks have been grouped according to the value taken by ν_1 . (b,d) they have been grouped according to the value taken by ν_2 . (c,d) CDF of the average value of k_{min} that minimizes D between the in-degree distribution of each interaction network and its best-fit power-law model.

the null hypothesis that such value is the same as the corresponding statistics in the control group, against the alternative hypothesis that the former is greater than the latter.

5 Discussion and conclusion

We started this work in the hope of discovering a simple statistical test that could be used to assess the presence and effectiveness of online community management policies, onboarding among them. Enacting onboarding on an online community leads to a strong rejection of a power-law behaviour hypothesis on its degree distribution. So, indeed, we can test for *the presence* of onboarding by looking at the degree distribution itself, which is much simpler than analysing the network's whole

Table 3: Average values of the power-law model’s exponent α in the control group and in the treatment group by values of v_1 and v_2 , computed over the whole support $k \geq 1$ (top) and $k \geq k_{min}$ (bottom). The number in parenthesis is the p-value associated to a t-test that $\alpha(treatment) = \alpha(control)$; they were omitted for $k \geq k_{min}$ as they are all smaller than 0.001.

$k \geq 1$ Control group: 1.752						
	$v_1 = 0.0$	$v_2 = 0.2$	$v_2 = 0.4$	$v_2 = 0.6$	$v_2 = 0.8$	$v_2 = 1$
$v_1 = 0.0$	1.89 (0.00)	1.89 (0.00)	1.89 (0.00)	1.89 (0.00)	1.89 (0.00)	1.89 (0.00)
$v_1 = 0.2$	1.85 (0.00)	1.85 (0.00)	1.85 (0.00)	1.85 (0.00)	1.85 (0.00)	1.85 (0.00)
$v_1 = 0.4$	1.82 (0.00)	1.82 (0.00)	1.82 (0.00)	1.82 (0.00)	1.82 (0.00)	1.82 (0.00)
$v_1 = 0.6$	1.79 (0.00)	1.79 (0.00)	1.79 (0.00)	1.79 (0.00)	1.79 (0.00)	1.79 (0.00)
$v_1 = 0.8$	1.77 (0.00)	1.77 (0.00)	1.77 (0.00)	1.77 (0.00)	1.77 (0.00)	1.77 (0.00)
$v_1 = 1$	1.75 (0.21)	1.75 (0.20)	1.75 (0.26)	1.75 (0.43)	1.75 (0.24)	1.75 (0.19)
$k \geq k_{min}$ Control group: 2.419						
	$v_2 = 0.0$	$v_2 = 0.2$	$v_2 = 0.4$	$v_2 = 0.6$	$v_2 = 0.8$	$v_2 = 1$
$v_1 = 0.0$	2.985	2.989	2.868	3.000	3.004	3.015
$v_1 = 0.2$	2.855	2.852	2.868	2.834	2.821	2.854
$v_1 = 0.4$	2.746	2.727	2.735	2.725	2.739	2.749
$v_1 = 0.6$	2.661	2.655	2.632	2.650	2.656	2.623
$v_1 = 0.8$	2.562	2.602	2.571	2.553	2.554	2.553
$v_1 = 1$	2.496	2.527	2.518	2.514	2.514	2.499

topology. However, we did not find a monotonic relationship between onboarding’s effectiveness and the distance of the resulting degree distribution from a pure power-law form. So, our simple test cannot tell the analyst *how effective* these policies are.

Our models incorporates two forces: preferential attachment and onboarding. The former is meant to represent the (emergent) rich-get-richer effect observed in many real-world social networks; the latter is meant to represent the (command-and-control) onboarding action of moderators and community managers. The former’s effect is known to lead to the emergence of an in-degree distribution that approximates a power-law model. The latter’s effect is more subtle, because it is in turn composed of two other effects. One consists in the direct action of the moderator, which always targets the newcomer; the other results of the consequences of a well-executed onboarding policy.

The direct action of the moderators creates edges pointing to nodes not selected by preferential attachment: this is definitional of onboarding, and of other online community management activities. What (non-moderator) participants in the online community do as a result of moderator activity is not as clear cut. In our simulation model, fully successful onboarding results in extra edges, some of which point to nodes selected by preferential attachment, others to nodes selected otherwise.

Also, onboarding only targets newcomers. As many online community management policies, it concerns weakly connected participants in the community: moderators have no need to engage with very active, strongly connected participants, who clearly need no help in getting a conversation going. By engaging weakly con-

nected participants, moderators hope to help some shy newcomers turn into active community members. Once this process is under way, moderators have no reason to continue to engage with the same individuals. In terms of our model, this means that newcomers, after having being onboarded, are going to receive new edges by preferential attachment only. It is therefore reasonable to expect that the degree distributions generated by our model display a heavy tail, with the frequency of highly connected nodes following a reasonable approximation of a power law. The overall result of onboarding, then, is an in-degree distribution with power-law behavior for high values of in-degree k and non-power law behavior for low (close to 1) values of k . This is indeed what we observe.

Non-preferential attachment selection of edge targets leads to a poorer fit of power-law models to the in-degree distributions where onboarding is present. This effect takes three forms. The first one is that, fitting a power-law model to the network's in-degree distribution and then running goodness-of-fit tests return a lower p-value than the p-value returned by the same test when onboarding is absent. The second effect is that the value of k that minimizes D between the best-fit power-law model and the observed data tends to be higher than without onboarding. The third one is that the scaling parameter of the best-fit power law tends to be higher with onboarding: onboarding makes the allocation of incoming edges more equal.

Our specification of the model accounts for an apparent paradox: the deviation of the observed networks' degree distributions from power-law behavior is greater when onboarding is ineffective than when it is effective. Ineffective onboarding only adds edges directly created by moderators, *none* of which are allocated across existing nodes by preferential attachment. As onboarding gets more effective, even more edges are added; some are allocated by preferential attachment, and drive the degree distribution back towards a pure power-law behavior. This paradoxical response may explain why our community responsiveness parameter v_2 does not appear to impact the shape of the in-degree distribution.

5.1 Future work

Modeling online community management means accounting for the interplay of bottom-up forces (like preferential attachment) with top-down ones (like onboarding policies). This weaving of emergence and design is precisely what we wish to investigate. There are three obvious directions in which we plan to expand the present model. The most obvious one is a systematic exploration of the parameter space, with the goal of assessing our results' robustness with respect to model specification.

A second direction for further research would be to attempt to make the model into a more realistic description of a real-world online community. Such an attempt would draw attention onto how some real-world phenomena, when incorporated in the model, influence its results. It would also carry the advantage of allowing online community management professional to more easily interact with the model and critique it. Several issues that could be investigated in this vein come to mind. For example, we could relax the assumption that the additional attractiveness parameter

A_s is identical for all nodes, allowing for different nodes in the network to attract incoming edges at different rates (a phenomenon known as multiscaling [4]). Secondly, we could introduce a relationship between out-degree and in-degree: this would reflect the fact that, in an online community, reaching out to others (which translates in increasing one's own out-degree in the interaction network) is a good way to get noticed and attract incoming comments (which translates in an increase in one's in-degree). Finally, we could work with other community management policies.

A third direction for further research would attempt to gauge the influence of onboarding and other community management policies on network topology by indicators other than the shape of its degree distribution, such as the presence of subcommunities.

Additionally, we wish to obtain and analyse more empirical data from real-world online communities with and without onboarding policies.

References

- [1] Barabasi, A.L.: The origin of bursts and heavy tails in human dynamics. *Nature* **435**(7039), 207–211 (2005)
- [2] Barabási, A.L., Albert, R.: Emergence of scaling in random networks. *science* **286**(5439), 509–512 (1999)
- [3] Barabási, A.L., Albert, R., Jeong, H.: Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications* **272**(1), 173–187 (1999)
- [4] Bianconi, G., Barabási, A.L.: Competition and multiscaling in evolving networks. *EPL (Europhysics Letters)* **54**(4), 436 (2001)
- [5] Borgatti, S.P., Mehra, A., Brass, D.J., Labianca, G.: Network analysis in the social sciences. *science* **323**(5916), 892–895 (2009)
- [6] Burt, R.S.: *Structural holes: The social structure of competition*. Harvard university press (2009)
- [7] Clauset, A., Shalizi, C.R., Newman, M.E.: Power-law distributions in empirical data. *SIAM review* **51**(4), 661–703 (2009)
- [8] De Liddo, A., Sándor, Á., Shum, S.B.: Contested collective intelligence: Rationale, technologies, and a human-machine annotation study. *Computer Supported Cooperative Work (CSCW)* **21**(4-5), 417–448 (2012)
- [9] Diplaris, S., Sonnenbichler, A., Kaczanowski, T., Mylonas, P., Scherp, A., Janik, M., Papadopoulou, S., Ovelgoenne, M., Kompatsiaris, Y.: Emerging, collective intelligence for personal, organisational and social use. In: *Next generation data technologies for collective computational intelligence*, pp. 527–573. Springer (2011)
- [10] Dorogovtsev, S.N., Mendes, J.F.: Evolution of networks. *Advances in physics* **51**(4), 1079–1187 (2002)
- [11] Hodas, N.O., Lerman, K.: The simple rules of social contagion. *Scientific reports* **4** (2014)
- [12] Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pp. 56–65. ACM (2007)
- [13] Kunegis, J., Blattner, M., Moser, C.: Preferential attachment in online networks: measurement and explanations. In: *Proceedings of the 5th Annual ACM Web Science Conference*, pp. 205–214. ACM (2013)
- [14] Laniado, D., Tasso, R., Volkovich, Y., Kaltenbrunner, A.: When the wikipedians talk: Network and tree structure of wikipedia discussion pages. In: *ICWSM (2011)*
- [15] Levy, P.: *Collective intelligence: Mankinds emerging world in cyberspace*. Cambridge, Mass.: Perseus Books (1997)

- [16] Lewis, K., Kaufman, J., Gonzalez, M., Wimmer, A., Christakis, N.: Tastes, ties, and time: A new social network dataset using facebook. com. *Social networks* **30**(4), 330–342 (2008)
- [17] Nick, B.: Toward a better understanding of evolving social networks. Ph.D. thesis (2013)
- [18] Rheingold, H.: *The virtual community: Homesteading on the electronic frontier*. MIT press (1993)
- [19] Shirky, C.: *Here comes everybody: The power of organizing without organizations*. Penguin (2008)
- [20] Shum, S.B.: The roots of computer supported argument visualization. In: *Visualizing argumentation*, pp. 3–24. Springer (2003)
- [21] Slegg, J.: Facebook news feed algorithm change reduces visibility of page updates (2014). URL <http://searchenginewatch.com/sew/news/2324814/facebook-news-feed-algorithm-tweak-reduces-visibility-of-page-updates>
- [22] Zanetti, M.S., Sarigol, E., Scholtes, I., Tessone, C.J., Schweitzer, F.: A quantitative study of social organisation in open source software communities. arXiv preprint arXiv:1208.4289 (2012)
- [23] Zhang, J., Ackerman, M.S., Adamic, L.: Expertise networks in online communities: structure and algorithms. In: *Proceedings of the 16th international conference on World Wide Web*, pp. 221–230. ACM (2007)

A Temporal-Causal Network Model for the Relation Between Religion and Human Empathy

Laila van Ments, Peter Roelofsma and Jan Treur

Abstract Religion has been extensively studied from many different perspectives. The current study aims at integrating a number of these perspectives into one computational network model. By first developing a conceptual temporal-causal network model based on literature, and then formalizing this model into a numerical network model, simulations can be done for almost any kind of religious person, showing different behaviours for persons with different religious backgrounds and characters. The focus was mainly on the influence of religion on human empathy and disempathy, a topic very relevant today.

1 Introduction

Religion is a topic that every person has an opinion about, whether that opinion is positive or negative. While some people blame religion for war and terrorism, others believe that religion is the only bright spot in a world full of bad. Does religion cause individuals to be more empathic, enabling them to be aware of the others feelings, needs and wants? Or, is religion a cause for human disempathy, making persons indifferent or even hostile for their fellow human? A clear answer has not yet been found, even though a lot of research has been done on the topic; e.g., [19, 21, 27, 37]. Questioning the influence of religion on human behaviour may not deserve a yes or a no type of answer, but rather an answer that involves more aspects, like ones character, culture, and of course different kinds of religions. In some way, all aspects and influences indicated above come together and originate in the brain. A lot of research has been done on how human behaviour is generated in the brain, also

Laila van Ments (e-mail: lailavanments@hotmail.com)✉ · Jan Treur (e-mail: treur@cs.vu.nl)✉

Behavioural Informatics Group, Vrije Universiteit Amsterdam, The Netherlands

Peter Roelofsma (e-mail: proelofsma@yahoo.co.uk)✉

Seminary for Migrant Churches and Theology, Vrije Universiteit Amsterdam

concerning religious topics. So, if these processes in the brain related to religion can be represented, this could help to get an answer to the question.

A method that can be used to represent real-world processes concerning human beings is Network-Oriented Modelling. By this method, mechanisms that are based on neurological mechanisms are represented in a network model using different states and connections between them, as described in [33, 34]. This Network-Oriented Modelling method can be used to simulate behaviour of individuals with different religious backgrounds, characters and cultures. In this paper, first, in Section 2 a brief literature overview on the existing research related to the topic is discussed. Then, in Section 3 the conceptual representation of the network model with its various parts is discussed, and it is indicated how a numerical formalization of this model was obtained. In Section 4 a relevant scenario simulated using the model is discussed; Section 5 is a discussion.

2 Literature Overview

There are two important approaches that are used to explain the origins of religion and religion-based behaviour. First, there is the evolutionarist approach [2, 7] that tries to explain the origin and different aspects of religion from an evolutionary perspective. Secondly, there is the neurotheologist approach [4, 8, 25] that tries to find the origins of religion in the brain and explain religious behaviour on the basis of neurological processes. Further scientific and philosophical developments from both different perspectives around cognition, neuroscience and conscious thinking will most likely generate useful insights into religion [37]. Therefore, an approach that combines these different aspects into one model would give the most promising answer to our question. Such a kind of multidisciplinary model is indicated in two articles by Kapogiannis et al. [18, 19], proposing an integrative cognitive neuroscience framework for understanding the cognitive and neural foundations of religions. Among others using MRI analysis, they define three dimensions that together form an individuals religious belief. The first one is Gods perceived level of involvement, the second Gods perceived level of emotion, and finally the doctrinal and experiential religious knowledge of an individual. Kapogiannis et al., considered these dimensions as nodes of a network and examined the causal flow within and between such networks, together forming the individuals religious belief. Also some other studies on religion have been combining knowledge from multiple disciplines, like [26, 38], although the distinction between the different perspectives on religion was still kept.

Besides the above described approaches to religion, many experiments have been done to examine behaviour of religious persons. As explored by [21], religion can foster implicit self-regulation among religious individuals, unconsciously changing their actions and regulating their emotions. Also, religious individuals that prayed for people that angered them showed less aggression towards those people afterwards, indicating that religious behaviour can change peoples emotions [6]. Furthermore, a study of Schjoedt et al. [29] found that praying towards God activates brain regions that are responsible for active interpersonal interactions and enable people to generate

an internal representation about the other, in this case God. This proves that praying individuals consider God a real meaningful person, rather than a fictive or abstract entity. This idea of internally representing God as a person is also discussed in [26].

Regarding this theory of God as a real meaningful person, an interesting idea can be developed as follows. As described in [22, 31, 32], a person can develop an empathic understanding of others through mirroring and internal simulation mechanisms, and these mechanisms also influence the individual beliefs and actions of that person. As a result, the aforementioned internal representation that individuals generate when they communicate with God, as a real meaningful person, can also generate an empathic understanding of God as perceived by the individual. This way, the individual mirrors the (internally represented) beliefs, actions and emotions of their perceived God. The combination of these mechanisms enables the image that an individual has of God to influence his own beliefs, actions and emotions, in a way similar to how an individual is influenced by other humans. The image that an individual has of God (e.g. the God-image which will be described more extensively later on), and how this image has impact on the individual, can involve many aspects. One example is studied by Granqvist et al. [14], who examined the God-image as an attachment figure in theistic religions, defining the relationship with God as an attachment relationship. Granqvist et al. examine the influence of a persons attachment style to the persons relationship to God. Another example that was studied is the impact of the character that an individuals God-image has. For example, an individual whose God-image is based on an authoritarian figure (like God is great, or God strikes down in anger) act in more antisocial, disempathic ways, and believers whose attachment relationship with God is a loving one (God is love) are acting in a more social, empathic manner [11, 17, 24]. Finally, there is an influence of the level of judgmentalism in a persons God image on the willingness to volunteer both in internal and external communities [23]. However, as described above, the influence of religion on human empathy and dis-empathy does not emerge from one single input, but from the combination of the individuals character and his God-image, which are both (partly) formed by the individuals experiences and knowledge.

3 The Temporal-Causal Network Model

In this section, it is presented how a neurologically inspired network model can be made that simulates the influence of religion on an individuals (dis)empathic behaviour and emotions towards others. The model was developed according to the Network-Oriented Modelling approach based on temporal-causal networks described in [33, 34] and adopts elements of previously developed network models for joint decision making processes [32] and action ownership [30]. It is based on different theories on religion and human behaviour from literature which will be explained below. Combining these elements, an integrative computational model was created that focuses on the influence of religion on (dis)empathic behaviour and emotions towards others. First, Sections 3.1, 3.2, and 3.3 present how theories and literature

were used to construct the model, leading to a conceptual representation of the network model depicted in Fig. 1. Then, Section 3.4 explains how a numerical representation was obtained from this conceptual representation.

3.1 *Mirror Neurons and Internal Simulation*

Mirror neurons enable sensory input, for example an observed action or body state of another person, to directly affect a persons own preparation state. In the current model, this is modelled by direct links from the sensory representation states of the emotions and actions of the God-image to the preparation states for emotions and behaviour of the Self. This gives the preparation state a similar function as a mirror neuron has: become active after observing the action or emotion. This mirror neuron function of preparation states makes that the actions and emotions of the God-image affect the corresponding behaviour, emotion and prayer states of the Self, leading to the actions and emotions of the God-image to influence the behaviour, emotions and prayers of the Self. The mirror neuron function enables to influence the individuals own preparation states. Then, due to activation of the preparation states, the actions or emotions are internally simulated in a process as described by William James [16] and Antonio Damasio [9, 10]; this involves the following process. A world state w_{SW} , a situation W in the world, occurs representing another persons action or emotion expression X .

The person develops a sensory state ss_W of this world state, and then a sensory representation state srs_W of it. Now by its mirror neuron function the preparation state ps_X for bodily changes for the same action or emotion X occurs. Depending on the context, this is expressed or executed, indicated by state es_X . Execution of an action is modelled by an *action execution loop* and the process involving expression of an emotion by a *body loop*. In the model, the body loop is modelled by the link from an individuals execution state of a body state expressing an emotion to the individuals sensory representation of that body state. The feeling for the emotion is based on this sensory representation of the body state. However, the process is extended by adding a possibility by internal simulation without executing an actual action (as-if body loop). This process, is incorporated in the model by a (predictive) loop from the preparation state for an action or emotion to the sensory representation for its effect, enabling direct emotion formation without behaviour execution.

3.2 *Action Ownership States of God and Self*

Whether an individual performs certain behaviour or expresses emotions that were mirrored (e.g., from the God-image) depends on the context. This context is represented by action ownership states for which a model was introduced in [30]. An ownership state is an indication to what extent an individual attributes an action or emotion to himself, or to what extent the individual deems someone else responsible. This ownership state for an action (which can also apply to an emotional response)

can lead to a go or no-go decision for behaviour or emotion expression. There are four different ownership states in the model; see Table 1.

Table 1: Ownership states for God and Self for actions and emotions.

	God-ownership state	Self-ownership state
action	OS_{God,a_i,e_i,b_i}	OS_{Self,c_i,e_i,b_i}
emotion	OS_{God,b_i,e_i}	OS_{Self,b_i,e_i}

Here OS_{Self,c_i,e_i,b_i} is the Self-ownership state for behaviour c_i with predicted effect e_i and related feeling b_i . It is influenced by the sensory representation state srs_{God,a_i} of God performing action a_i , the sensory representation of e_i and the feeling state for b_i . In turn, it influences both the preparation state ps_{c_i} for that behaviour c_i and the execution state es_{c_i} for that behaviour. Furthermore, God-ownership OS_{God,a_i,e_i,b_i} for action a_i is influenced by the sensory representation states srs_{God,a_i} and $srs_{God,image_i}$ of the God-image. In turn, by mirroring it affects the preparation state ps_{c_i} for the related behaviour of Self and the execution state es_{c_i} for that behaviour. Moreover, Self-ownership OS_{Self,b_i,e_i} of emotional response b_i related to e_i is influenced by the sensory representation srs_{God,b_i} of the emotion within the God-image and the persons sensory representation srs_{e_i} of the predicted effect e_i . In turn, OS_{Self,b_i,e_i} influences the preparation state ps_{b_i} of the emotion and the execution state es_{b_i} (expression) of the emotion b_i . Finally, God-ownership state OS_{God,b_i,e_i} of emotional response b_i related to e_i is influenced by the sensory representation states srs_{God,b_i} of Gods emotion and $srs_{God,image_i}$ of the God-image. In turn, it affects the preparation state ps_{d_i} for the related emotion d_i and the execution state es_{d_i} for that emotion. With the distinction between the ownership of God over behaviour and emotions that the individual expresses, the level of involvement and authority of God that an individual experiences is represented, as brought forward in [18]. An individual with a very low Self-ownership and a high God-ownership can show behaviour different from an individual with a high Self-ownership and a low God ownership.

3.3 The God-image

The notion of the God-image has received a lot of attention in the scientific world in the past years, studying the influence of this phenomenon, and, more specifically its influence on human behaviour towards others [17, 23, 24]. Different kinds of God-images have proved to influence human behaviour towards others in different ways. For example, where an authoritarian, punishing and controlling God-image is correlated to aggressive, disempathic behaviour, a forgiving, helping God-images correlates to prosocial, empathic behaviour [17, 23]. Furthermore, the belief in Godly omnipresence and omnipotence also influences human prosociality: individuals

with a moralistic, all knowing God-image showed more prosocial behaviour than individuals with a non-moral or non-all-knowing God-image [24].

Besides the studies on the influence of the God-image on human behaviour towards others, this process can also be described from the mentalizing perspective, as introduced by Schaap-Jonker [26]. Mentalizing is the capacity of thinking about thinking and feeling. It provides awareness that ones own and others behaviour is driven by mental states, and gives the ability to selectively activate internal states that fit the individuals particular. Also, mentalizing generates a subjective experience of agency, this way supporting a sense of identity [1, 3, 12, 26]. Mentalizing also bears some resemblance to the process of internal simulation as described in [32], where an individual internally simulates mind states to predict effects in the external world or other persons. Mentalizing can occur both consciously or unconsciously, concern the self or others, and is both cognitive and affective [12]. This creates many possibilities in the interactions of the individual towards the God-image.

To enable a God-image to influence an individuals behaviour as explained above, the individual first has to have a God-image. The God-image refers to the personal God of the individual. As discussed in [18, 19, 27], this God-image consists of both an emotional part and a cognitive part, and both parts are dynamically interrelated. The emotional part is unconsciously developed, highly influenced by parents and significant others. The cognitive part of the God-image consists of the knowledge an individual has about God, like the doctrinal information the individual received in religious study, at school, or at church. The emotional and the cognitive part that form the God-image can be traced back to different parts in the brain as studied by [18, 19, 27]. The emotional part involves the amygdala, basal ganglia, the ventromedial prefrontal cortex, the lateral temporal cortex, the dorsal anterior cingulated cor-tex and the orbitofrontal cortex. These parts of the brain are involved in assigning emotional significance to behaviour and events and to controlling cognition and emotion. On the other hand, the cognitive part involves the lateral prefrontal cortex, the medial prefrontal cortex, the lateral parietal cortex, the medial parietal cortex and the medial temporal lobe, all brain circuits that are responsible for the processing of more complex linguistic and symbolic input. This combination of brain processes results in the formation of the personal God-image of the individual; each personal God-image differs based on the individuals personal character, experiences and knowledge, which will be discussed more extensively below.

To summarize, both the doctrinal knowledge that an individual receives about God, and the individuals character, upbringing and so forth, create a personal, internal God-image that the individual perceives as a real person, and with whom the individual interacts. In the computation model, the God-image is represented by the following process. The generation of the God-image happens through the links between the external input (World states) to the sensor states, and in the links from the sensor states to the sensory representations of the God-image. Then, the God-image influences the behaviour and emotions of the individual through the links from the sensory representations of the God-image to the ownership states, goal fulfilment state, and the preparation states.

As described above, the individual imagines God as a person with intentions and mind states [29]. In the developed model, the God-image (including images of Gods actions and emotions) is constructed by three different kinds of input, namely input about Gods emotions (mind states), actions that God performs (or intentions), and about the God image in general. This input can come from many sources, for example religious texts or education from parents, or from prayer. The generation of the God-image from the input is modelled by the links from the world states to the sensor states (including the sensor state of the prayer, representing hearing of a prayer of someone else or of oneself), and from the sensor states to the sensory representation states of the (general) God-image, God actions and God emotions. Furthermore, while an individuals own prayer can influence the God-image via an external connection, the individuals prayer can also influence that individuals God-image via an internal connection, based on links from the preparation state for the prayer to the sensory representations states for actions and emotions of the God-image and the general God-image; e.g. if an individual prays to make God happy, the emotion of his God-image might become happier (depending on the individuals beliefs). Part of the God-image is represented by the (adaptive) connection weights within the God-image model, partly representing the individuals characteristics, and which may be influenced by the external input as well through Hebbian learning. These parts result in a personal God-image consisting of the individuals sensory representation of the God-image, the individuals sensory representation of Gods actions, the individuals sensory representation of Gods emotions, and the weights of the connections between these three states. The conceptual representation of the model is graphically depicted in Fig. 1. In this representation, circles represent states and arrows represent processes. The dotted arrows represent Hebbian learning connections, which will be explained below. The processes that are internal are depicted inside the green box, external processes are outside the box, and the interaction between the two on the boundary.

The subscript i represents the difference between empathetic and disempathic behaviour and emotion. An overview of the connections (the arrows) and their weights that were defined for the model, can be found in Appendix E in [36].

3.4 From Conceptual to Numerical Representation of the Model

This section describes the process of numerical formalization of the model presented in Sections 3.1 to 3.3. This formalization was used to implement the model in Python in order to perform simulations. According to the adopted Network-Oriented Modelling approach, a graphical conceptual representation displays nodes for states and arrows for connections indicating causal impacts from one state to another, and includes some additional labels for states and connections, so that it becomes a labeled graph:

- connection weights $\omega_{X,Y}$ for each connection from state X to state Y
- combination functions $c_Y(\dots)$ to aggregate multiple impacts for each state Y
- speed factors η_Y for speed of change for each state Y

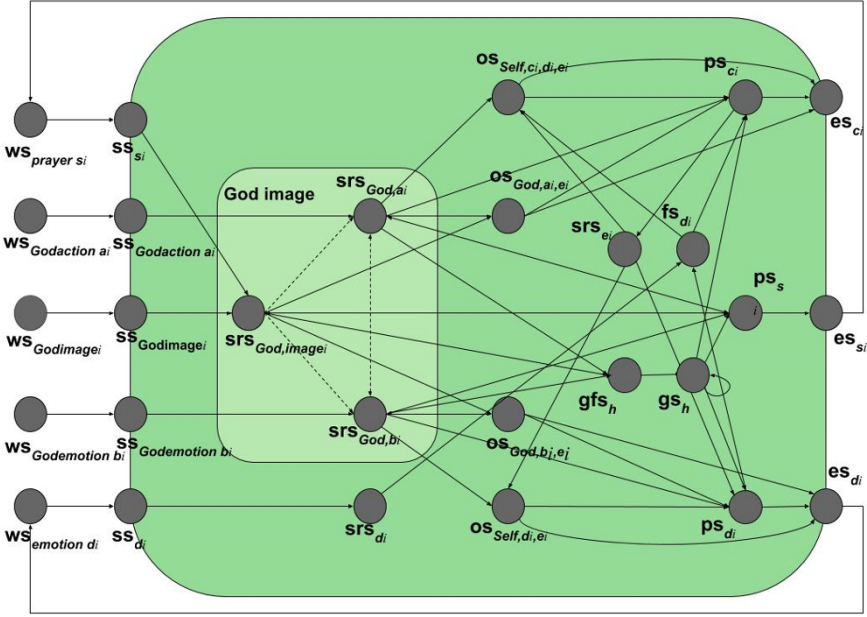


Fig. 1: Graphical conceptual representation of the temporal-causal network model; here subscript i denotes either empathy (1) or disempathy (2).

To choose combination functions, a number of standard options is available; e.g., [33, 34]. The conceptual representation of a temporal-causal network model can be transformed in a systematic or even automated manner into the following numerical representation of the model [33, 34]; here the variable t indicates a time point; it varies over the real numbers. Based on a combination function and the connection weights

$$\mathbf{aggimpact}_Y(t) = \mathbf{c}_Y(\boldsymbol{\omega}_{X_1, Y}, \dots, \boldsymbol{\omega}_{X_k, Y}(t)) \quad (1)$$

is the *aggregated impact* of the network on Y at t . This is used to provide the following *difference* and *differential equation* for each state Y :

$$\begin{aligned} Y(t + \Delta t) &= Y(t) + \boldsymbol{\eta}_Y[\mathbf{aggimpact}_Y(t) - Y(t)]\Delta t \\ &= Y(t) + \boldsymbol{\eta}_Y[\mathbf{c}_Y(\boldsymbol{\omega}_{X_1, Y}, \dots, \boldsymbol{\omega}_{X_k, Y}(t)) - Y(t)]\Delta t \end{aligned} \quad (2)$$

$$dY(t)/dt = \boldsymbol{\eta}[\mathbf{aggimpact}_Y(t) - Y(t)] = \boldsymbol{\eta}[\mathbf{c}_Y(\boldsymbol{\omega}_{X_1, Y}, \dots, \boldsymbol{\omega}_{X_k, Y}(t)) - Y(t)] \quad (3)$$

These numerical representations (2) and (3) can be used for mathematical and computational analysis and simulation. In the model presented here, for all states for the combination function the *advanced logistic sum combination function* $\mathbf{alogistic}_{\sigma, \tau}(\dots)$ is used [33, 34]:

$$c_Y(V_1, \dots, V_k) = \mathbf{alogistic}_{\sigma, \tau}(V_1, \dots, V_k) = \left(\frac{1}{1 + e^{-\sigma(V_1 + \dots + V_k - \tau)}} - \frac{1}{1 + e^{\sigma\tau}} \right) \quad (4)$$

Here σ is a *steepness* parameter and τ a *threshold* parameter. The advanced logistic sum combination function (4) has the property that activation levels 0 are mapped to 0 and it keeps values below 1. When the value of the right hand side expression given above is < 0 , the value 0 is assigned to $\mathbf{alogistic}_{\sigma, \tau}(V_1, \dots, V_k)$.

In cases of adaptive networks in which some or all of the connection weights $\omega_{X,Y}$ are dynamic, for a numerical representations dynamic connection weights also get a time argument: $\omega_{X,Y}(t)$. To model their dynamics, the dynamic connection weights are described by a difference or differential equation for Hebbian learning, which also can be based on a combination function and speed factor as above; for more details, see [33, 34]. In the current network model learning mechanism were included for the connection strengths of the adaptive connections from $srs_{God, image_i}$ to srs_{God, a_i} , from $srs_{God, image_i}$ to srs_{God, b_i} , from srs_{God, a_i} to srs_{God, b_i} , and from srs_{God, b_i} to srs_{God, a_i} ; see the dotted lines in Figure. 1. This learning mechanism is based on the Hebbian learning principle introduced by Donald Hebb [15]. Different interpretations of Hebbian learning exist, either based on causality-based learning [20] or simultaneity-based learning; e.g., [5, 13, 35]. In this model, the latter simultaneitybased learning approach is used. This approach is based on the principle that strengthening of a connection between neurons over time may take place when both nodes are often active simultaneously: neurons that fire together, wire together [28]. In the model, the weight $\omega_{X,T}$ of an adaptive connection from state X to state Y is updated after time step Δt using a learning rate $\eta_H > 0$ and extinction $\zeta_H \geq 0$ and the activation levels $X(t)$ and $Y(t)$ of the states X and Y . This is modelled as follows (see also [13], p. 406):

$$\omega_{X,Y}(t + \Delta t) = \omega_{X,Y}(t) + [\eta_H X(t)Y(t)(1 - \omega_{X,Y}(t)) - \zeta_H \omega_{X,Y}(t)]\Delta t \quad (5)$$

The weight $\omega_{X,Y}$ has a maximal strength of 1; the factor $1 - \omega_{X,Y}(t)$ keeps $\omega_{X,Y}$ below 1.

4 Simulation Scenario: a Person with Fundamentalist Tendencies

As discussed, the computational model was implemented in Python in order to perform simulations and study the influence of religion on human empathy and disempathy. Simulations have focused on six possible scenarios based on literature. All of them can be found in Appendix D in [36]. In the current section, for the sake of space limitations, only one of them is discussed. For each scenario, relevant parameter values are chosen in order to simulate the behaviour described in literature and to test the influence on empathic or disempathic behaviour. For most of the states in the implemented model two instances are used: the empathic (indicated with subscript 1 in the figures) and the disempathic instance (indicated with subscript 2 in the figures). Through adapting the connections relating to those two instances, the degree of empathy of disempathy of the God-image or individual can be varied. For each

scenario Δt was chosen 0.25, the total number of time steps 500, and the speed factor of all states 0.17. The extinction and learning rates for the adaptive connections are all 0.5. A certain combination of parameters within a person could lead to fundamentalist tendencies. If a person has both an anxious attachment relationship with the God image, a disempathic God-image, and a lot of divinity and disempathic related external influence about God, this could form behaviour that is considered fundamentalist.

This scenario aims at simulating this fundamentalist behaviour by making the disempathic connection weights in the model higher than the empathic ones (1.0 versus 0.1), making the connections for the God ownership states higher than the Self-ownership states (God-ownership for empathic behaviour become 0.8, for disempathic behaviour 0.3, Self-ownership 0.1) and strong links from a disempathic God-image to the preparation states (1.0) and from preparation states to execution states (1.0). The results can be found in Fig. 2.

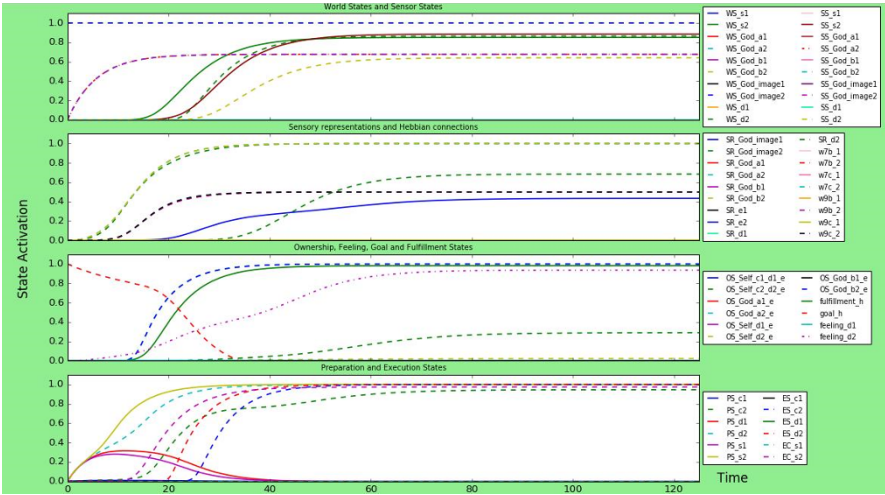


Fig. 2: Simulation scenario for a person with fundamentalist tendencies, meaning that connection strengths to the Self-ownership are very low while the God-ownership is high, there is low, connection strengths to the effect prediction is low and the person has a disempathic God-image. The person strongly executes disempathic behaviour, no empathic behaviour, and develops no Self-ownership.

Main differences with a scenario with a person with a disempathic God-image are as follows: the fundamentalist person does not, or barely, develop Self-ownership of its actions; the fundamentalist person does have a lower activation level of the prediction of the effects of his actions: srs_{e_i} . The disempathic behaviour of the fundamentalist person reaches the same activation level, but reaches this level faster than the person with just a disempathic God-image.

5 Discussion and Conclusion

In this paper the influence of religion on human empathy and disempathy was studied. First of all, an extensive literature study was done regarding all the processes are related to religion and human behaviour, specifically towards others. The relevant theory was then used to design a conceptual representation of a temporal-causal network model that captures the process of how religion influences human behaviour, for example the religion-related external input that an individual receives, the way this external input is then processes and generates a personal God-image, and how this God-image influences the individuals behaviour and emotions. The behaviour and emotions of both the God-image and the individual were distinguished in empathic and disempathic. Although (informally expressed) theories exist and are referred in the different sections above, a formalised computational model for them was never designed, as far as the authors know; so, comparison with other computational models is difficult.

The developed conceptual representation was then formalized into the numerical representation and this was implemented in Python. With this implemented network model, scenarios based on relevant literature were addressed to simulate the influence of religion on human empathy and disempathy, in order to answer the question asked in the beginning. For example, scenarios were simulated for a person with an empathic or disempathic God-image, persons with atheist or fundamentalist tendencies, or persons with Autism Spectrum Disorders. It was shown how a person mirrors the empathy or disempathy in the actions and emotions of the God-image, depending on the situation of a person. First of all, it was shown how external (religious) influences have impact on an individuals God-image. Input regarding a disempathic God created a disempathic God-image, while input regarding an empathic God gen-erated an empathic God-image. Furthermore, the God-image strongly influenced the empathic or disempathic behaviour and emotions of the religious individual. An em-pathic God-image led to empathic actions and emotions, while a disempathic God-image led to disempathic actions and emotions. However, there were more aspects that influenced this. For example, the ownership and mirroring process: persons with a very low Self-ownership can show more fundamentalist tendencies.

Although the simulations and the model in general show some interesting results, it is difficult to provide a final answer on what the influence of religion on human empathy and disempathy is. While the model does represent important aspects of the domain, and is a good basis for an answer, there are still many things to improve. For example, the model only reflected the influence of the God-image on the behaviour of the individual, not that of other persons or more specific non-addressed characteristics of the person itself. Therefore, the process of literature study, developing a conceptual model, formalizing it and simulating is an iterative one, where adaptations can be made all the time in order to match the real world situation as much as possible while preserving the abstractness that is required of a computational model.

References

- [1] Allen, J.G.: Mentalizing in practice. *Handbook of mentalization-based treatment* pp. 3–30 (2006)
- [2] Atran, S.: *In Gods we trust: The evolutionary landscape of religion*. Oxford University Press (2004)
- [3] Bateman, A.W., Fonagy, P.: *Handbook of mentalizing in mental health practice*. American Psychiatric Pub (2012)
- [4] Beauregard, M., Paquette, V.: Neural correlates of a mystical experience in carmelite nuns. *Neuroscience letters* **405**(3), 186–190 (2006)
- [5] Beste, C., Dinse, H.: Learning without training. *Current Biology* **23**(11), R489 – R499 (2013). DOI <http://dx.doi.org/10.1016/j.cub.2013.04.044>. URL <http://www.sciencedirect.com/science/article/pii/S0960982213004855>
- [6] Bremner, R.H., Koole, S.L., Bushman, B.J.: pray for those who mistreat you: Effects of prayer on anger and aggression. *Personality and Social Psychology Bulletin* **37**(6), 830–837 (2011)
- [7] Bulbulia, J.: The cognitive and evolutionary psychology of religion. *Biology and philosophy* **19**(5), 655–686 (2004)
- [8] Cooke, P., Elcoro, M.: Neurotheology: Neuroscience of the soul. *Journal of Young Investigators* **25**(3), 1–6 (2013)
- [9] Damasio, A.: *Self Comes to Mind: Constructing the Conscious Brain*. Pantheon Books (2010). URL <https://books.google.it/books?id=47718f0rHoIC>
- [10] Damasio, A.R.: Descartes' error: Emotion, reason, and the human brain (1994)
- [11] Escher, D.: How does religion promote forgiveness? linking beliefs, orientations, and practices. *Journal for the Scientific Study of Religion* **52**(1), 100–119 (2013)
- [12] Fonagy, P.: *The Mentalization-Focused Approach to Social Development*, pp. 51–99. John Wiley & Sons, Ltd (2008)
- [13] Gerstner, W., Kistler, W.M.: Mathematical formulations of hebbian learning. *Biological cybernetics* **87**(5–6), 404–415 (2002)
- [14] Granqvist, P., Mikulincer, M., Shaver, P.R.: Religion as attachment: Normative processes and individual differences. *Personality and Social Psychology Review* **14**(1), 49–59 (2010)
- [15] Hebb, D.O.: *The organization of behavior* (1949)
- [16] James, W.: What is an emotion? *Mind* **9**(34), 188–205 (1884)
- [17] Johnson, K.A., Li, Y.J., Cohen, A.B., Okun, M.A.: Friends in high places: The influence of authoritarian and benevolent god-concepts on social attitudes and behaviors. *Psychology of Religion and Spirituality* **5**(1), 15 (2013)
- [18] Kapogiannis, D., Barbey, A.K., Su, M., Zamboni, G., Krueger, F., Grafman, J.: Cognitive and neural foundations of religious belief. *Proceedings of the National Academy of Sciences* **106**(12), 4876–4881 (2009)
- [19] Kapogiannis, D., Deshpande, G., Krueger, F., Thornburg, M.P., Grafman, J.H.: Brain networks shaping religious belief. *Brain connectivity* **4**(1), 70–79 (2014)
- [20] Keysers, C., Gazzola, V.: Hebbian learning and predictive mirror neurons for actions, sensations and emotions. *Phil. Trans. R. Soc. B* **369**(1644), 20130,175 (2014)
- [21] Koole, S.L., McCullough, M.E., Kuhl, J., Roelofsma, P.H.M.P.: Why religions burdens are light: From religiosity to implicit self-regulation. *Personality and Social Psychology Review* **14**(1), 95–107 (2010)
- [22] Memon, Z.A., Treur, J.: An agent model for cognitive and affective empathic understanding of other agents. In: *Transactions on Computational Collective Intelligence VI*, pp. 56–83. Springer (2012)
- [23] Mencken, F.C., Fitz, B.: Image of god and community volunteering among religious adherents in the united states. *Review of Religious Research* **55**(3), 491–508 (2013)
- [24] Purzycki, B.G., Apicella, C., Atkinson, Q.D., Cohen, E., McNamara, R.A., Willard, A.K., Xygalatas, D., Norenzayan, A., Henrich, J.: Moralistic gods, supernatural punishment and the expansion of human sociality. *Nature* (2016)

- [25] Sayadmansour, A.: Neurotheology: The relationship between brain and religion. *Iranian Journal of Neurology* **13**(1), 52–55 (2014)
- [26] Schaap-Jonker, H., Corveleyn, J.M.T.: Mentalizing and religion. *Archive for the Psychology of Religion* **36**(3), 303–322 (2014)
- [27] Schaap-Jonker, H., Sizoo, B., van Schothorst-van Roekel, J., Corveleyn, J.: Autism spectrum disorders and the image of god as a core aspect of religiousness. *The International Journal for the Psychology of Religion* **23**(2), 145–160 (2013)
- [28] Schatz, C.J.: The developing brain. *Scientific American* **267**(3), 60–67 (1992)
- [29] Schjoedt, U., Stødkilde-Jørgensen, H., Geertz, A.W., Roepstorff, A.: Highly religious participants recruit areas of social cognition in personal prayer. *Social Cognitive and Affective Neuroscience* **4**(2), 199–207 (2009)
- [30] Treur, J.: A cognitive agent model incorporating prior and retrospective ownership states for actions. In: *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, pp. 1743–1749 (2011)
- [31] Treur, J.: From mirroring to the emergence of shared understanding and collective power. In: *International Conference on Computational Collective Intelligence*, pp. 1–16 (2011)
- [32] Treur, J.: Modelling joint decision making processes involving emotion-related valuing and empathic understanding. In: *International Conference on Principles and Practice of Multi-Agent Systems*, pp. 410–423. Springer (2011)
- [33] Treur, J.: Dynamic modeling based on a temporal–causal network modeling approach. *Biologically Inspired Cognitive Architectures* **16**, 131–168 (2016)
- [34] Treur, J.: Network-Oriented Modeling: Addressing Complexity of Cognitive, Affective and Social Interactions. *Understanding Complex Systems*. Springer International Publishing (2016). URL <https://books.google.it/books?id=LcowDQAAQBAJ>
- [35] Treur, J., Umair, M.: Emotions as a vehicle for rationality: Rational decision making models based on emotion-related valuing and hebbian learning. *Biologically Inspired Cognitive Architectures* **14**, 40 – 56 (2015)
- [36] URL: Appendices. <http://few.vu.nl/~treur/religionappendices.pdf>
- [37] Visala, A.: *Cognition, Brain, and Religious Experience: A Critical Analysis*, pp. 1553–1568. Springer Netherlands, Dordrecht (2015)
- [38] Weingarten, C.P., Luborsky, L., Andrusyna, T., Diguier, L., Descôteaux, J.: Relationships between god and people: An interpersonal study of scriptures. *International Journal for the Psychology of Religion* **24**(2), 133–150 (2014)

Network-Oriented Modeling and Analysis of Dynamics Based on Adaptive Temporal-Causal Networks

Jan Treur

Abstract This paper discusses how Network-Oriented Modelling based on adaptive temporal-causal networks can be used to model and analyse dynamics and adaptivity of various processes. Adaptive temporal-causal network models incorporate a dynamic perspective on causal relations in which the states in the network change over time due to the causal relations, and these causal relations themselves also change over time. It is discussed how modelling and analysis of the dynamics of the behaviour of these network models can be performed.

1 Introduction

Network-Oriented Modelling has been proposed as a modeling perspective suitable for processes that are highly dynamic, circular and interactive; e.g., [26, 27]. In different application areas this modelling perspective has been proposed in different forms: in the context of modelling organisations and social systems (e.g., [3, 7, 20]), of modelling metabolic processes (e.g., [4]), and of modelling electromagnetic systems (e.g., [8, 9, 23]). To address dynamics well, Network-Oriented Modeling based on adaptive temporal-causal networks has been developed [25, 26, 27]. This approach incorporates a continuous (real) time dimension. Adaptive temporal-causal network models are dynamic in two ways: their states change over time based on the causal relations in the network, but these causal relations may also change over time. As in such networks often many interrelating cycles occur, their emerging behaviour patterns are not always easy to predict or analyse. This may make it hard to evaluate whether observed outcomes of simulations are plausible or might be due to implementation errors.

However, some specific types of properties can also be analysed by calculations in a mathematical manner, without performing simulations; e.g., [2, 17, 18, 19, 21, 22].

Jan Treur (e-mail: treur@cs.vu.nl)✉

Behavioural Informatics Group, Vrije Universiteit Amsterdam, The Netherlands

Such properties that are found in an analytic mathematical manner can be used for verification of the model by checking them for the values observed in simulation experiments. If one of these properties is not fulfilled (and the mathematical analysis was done in a correct manner), then there will be some error in the implementation of the model. In this paper methods to analyse such properties of temporal-causal network models will be described. They will be illustrated for two types of adaptive temporal-causal network models: one based on Hebbian learning (Section 3), and one based on the homophily principle for dynamic connection weights in adaptive networks modelling social interaction (Section 4).

2 Network-Oriented Modeling by Temporal-Causal Networks

The Network-Oriented Modeling approach based on temporal-causal networks described in more detail in [25, 26] is a generic and declarative dynamic modeling approach based on networks of causal relations. Dynamics is addressed by incorporating a continuous time dimension. This temporal dimension enables modelling by networks that inherently contain cycles, such as networks modeling mental or brain processes, or social interaction processes, and also enables to address the timing of the processes in a differentiated manner. The modeling perspective can incorporate ingredients from different modeling approaches, for example, ingredients that are sometimes used in neural network models, and ingredients that are sometimes used in probabilistic or possibilistic modeling. It is more generic than such methods in the sense that a much wider variety of modeling elements are provided, enabling the modeling of many types of dynamical systems, as described in [25, 26]. The Network-Oriented Modeling approach is supported by a few modeling environments (in Matlab, or in Python, for example) that can be used to model conceptually in a declarative manner, without the need of programming.

Temporal-causal network models can be represented at two levels: by a conceptual representation and by a numerical representation. A conceptual representation of a temporal-causal network model can have a (labeled) graphical form (or a matrix form), as shown in the examples presented below. In the first place it involves representing in a declarative manner states and connections between them. The connections represent (causal) impacts of states on each other, as assumed to hold for the application domain addressed. Each state X is assumed to have an (activation) level that varies over time, indicated in the numerical representation by a real number $X(t)$. In reality not all causal relations are equally strong, so some notion of strength of a connection from a state X to a state Y is used: a *connection weight* $\omega_{X,Y}$. Based on this, in a numerical representation the *impact* of state X on state Y at time t is defined by $\omega_{X,Y}X(t)$, where $X(t)$ is the activation level of state X at t . Note that also a connection from a state Y to itself is allowed. The weight $\omega_{Y,Y}$ of such a connection can, for example, be used to model persistence of state Y . Furthermore, when more than one causal relation affects a given state Y , these causal effects have to be combined. To this end, some way to *aggregate multiple causal impacts* on a state is used; this is done by a *combination function* $c_Y(\dots)$ that uses the impacts

$\omega_{X_i,Y}X_i(t)$ from states X_1, \dots, X_k on Y as input and provides one *aggregated impact* value out of them. Moreover, not every state has the same extent of flexibility in responding to impact; some states respond fast, and other states may be more rigid and may respond more slowly. Therefore, a *speed factor* η_Y of a state Y is used for timing of effectuation of causal impacts.

Combination functions can have different forms. The applicability of a specific combination rule may depend much on the type of application addressed, and even on the type of states within an application. Therefore, for the Network-Oriented Modeling approach based on temporal-causal networks a number of standard combination functions are available as options and a number of relevant properties of such combination functions have been identified; e.g., see [25], Table 10, or [26], Chapter 2, Table 2.10. Some of these standard combination functions are scaled sum, product, complementary product, max, min, and simple and advanced logistic sum functions. These options cover elements from different existing approaches, varying from approaches considered for reasoning with uncertainty, probability, possibility or vagueness, to approaches based on neural networks; e.g. [1, 5, 6, 10, 12, 14, 15, 16, 29]. In addition, there is still the option to specify any other (non-standard) combination function.

The above three concepts (connection weight, combination function, speed factor) can be considered as parameters representing characteristics in a network model. In a non-adaptive network model these parameters are fixed over time. But to model processes by adaptive networks, not only the state levels, but also these parameters can change over time. For example, the connection weights can change over time to model evolving connections in network models. For modeling processes as adaptive networks, some of the parameters (such as connection weights) are handled in a similar manner as states. For more detailed explanation, see below in Section 3.

A conceptual representation of a temporal-causal network model can be transformed in a systematic and automated manner into a numerical representation of the model, as described in [25, 26], thus obtaining the following *difference* and *differential equation* for all states Y :

$$Y(t + \Delta t) = Y(t) + \eta_Y [c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) - Y(t)]\Delta t \quad (1)$$

$$dY(t)/dt = \eta_Y [c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) - Y(t)] \quad (2)$$

The modeling approach enables to take into account theories and findings from any domain from, for example, biological, psychological, neurological or social sciences, as such theories and findings are often formulated in terms of causal relations. This applies, among others, to mental processes based on complex brain processes, which, for example, often involve dynamics based on interrelating and adaptive cycles. But equally well it applies to social interaction processes and their adaptive dynamics. This enables to address complex adaptive phenomena such as the integration of emotions within all kinds of cognitive processes, of internal simulation and mirroring of mental processes of others, and dynamic social interaction patterns, as shown in [26] by a large number of example models.

3 Modelling Mental Processes by Adaptive Networks

Mental processes can be modeled by temporal-causal networks in an adaptive manner: characteristics represented by network parameters can change over time as well. These parameters that can change are modeled in the same way as states. This will be illustrated here for one specific case: the way in which connection strengths can change based on Hebbian learning. In Section 4 a similar type of adaptivity will be illustrated for adaptive network models for evolving social interactions.

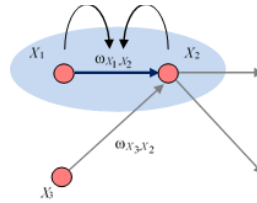
Hebbian learning [13], is based on the principle that strengthening of a connection between neurons over time may take place when both states are often active simultaneously (neurons that fire together, wire together); see also Fig. 1. The principle itself goes back to Hebb [13], but see also, e.g., [11]. In the example model considered here it is assumed that the strength ω_{X_1, X_2} of the connection from state X_1 to state X_2 is adapted using the following *Hebbian learning rule*, taking into account a maximal connection strength 1, a *learning rate* $\eta > 0$ and a *persistence factor* μ in the interval $[0, 1]$, and activation levels $X_1(t)$ and $X_2(t)$ (assumed between 0 and 1) of the two states involved:

$$d\omega_{X_1, X_2}(t)/dt = \eta[X_1(t)X_2(t)(1 - \omega_{X_1, X_2}(t)) - (1 - \mu)\omega_{X_1, X_2}(t)] \quad (3)$$

$$d\omega_{X_1, X_2}(t + \Delta t) = \omega_{X_1, X_2}(t) + \eta[X_1(t)X_2(t)(1 - \omega_{X_1, X_2}(t)) - (1 - \mu)\omega_{X_1, X_2}(t)]\Delta t \quad (4)$$

Such Hebbian learning rules can be found, for example, in (Gerstner and Kistler,

Fig. 1: Graphical conceptual representation of an adaptive network for Hebbian learning.



2002, p. 406). It will be discussed how this can be modeled by considering the connection weight ω_{X_1, X_2} as a state Ω_{X_1, X_2} that changes over time, represented by an extra node in the network. As a first step this node for the state Ω_{X_1, X_2} representing ω_{X_1, X_2} is added and connected; see Fig. 2 for a conceptual representation. This state is affected by both X_1 and X_2 due to the learning, so connections from these states to Ω_{X_1, X_2} are incorporated. Moreover a connection from Ω_{X_1, X_2} to X_2 is used to represent the effect of the connection strength on X_2 , and a connection from Ω_{X_1, X_2} to itself for persistence. The weights of all these connections are assumed 1; see Fig. 2. As a next step it is explored what combination functions are needed for Ω_{X_1, X_2} and X_2 in this new situation depicted in Fig. 2.

First, the combination function for the state Ω_{X_1, X_2} is identified, to aggregate the impacts of X_1 and X_2 , and Ω_{X_1, X_2} on Ω_{X_1, X_2} . The difference equation for the connection weight ω_{X_1, X_2} shown in (4) above can be rewritten into:

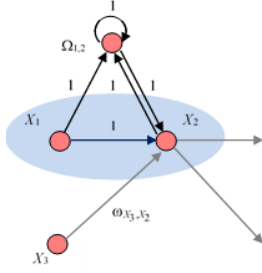


Fig. 2: Graphical conceptual representation for the Hebbian learning principle with state Ω_{X_1,X_2} representing a dynamic connection weight ω_{X_1,X_2} .

$$\begin{aligned}\Omega_{X_1,X_2}(t + \Delta t) &= \Omega_{X_1,X_2}(t) + \boldsymbol{\eta}[X_1(t)X_2(t)(1 - \Omega_{X_1,X_2}(t)) - (1 - \boldsymbol{\mu})\Omega_{X_1,X_2}(t)]\Delta t \\ &= \Omega_{X_1,X_2}(t) + \boldsymbol{\eta}[X_1(t)X_2(t)(1 - \Omega_{X_1,X_2}(t)) + \boldsymbol{\mu}\Omega_{X_1,X_2}(t) - \Omega_{X_1,X_2}(t)]\Delta t\end{aligned}\quad (5)$$

On the other hand, according to the temporal-causal network approach using a combination function $\mathbf{c}_{\Omega_{X_1,X_2}}(\dots)$ for state Ω_{X_1,X_2} (see equation (1)) it holds:

$$\Omega_{X_1,X_2}(t + \Delta t) = \Omega_{X_1,X_2}(t) + \boldsymbol{\eta}_{\Omega_{X_1,X_2}}[\mathbf{c}_{\Omega_{X_1,X_2}}(X_1(t), X_2(t), \Omega_{X_1,X_2}(t)) - \Omega_{X_1,X_2}(t)]\Delta t\quad (6)$$

So, the speed factor $\boldsymbol{\eta}_{\Omega_{X_1,X_2}}$ can be assumed $\boldsymbol{\eta}$, and it follows from equations (5) and (6) that the combination function $\mathbf{c}_{\Omega_{X_1,X_2}}(V_1, V_2, W)$ for the new state Ω_{X_1,X_2} satisfies

$$\mathbf{c}_{\Omega_{X_1,X_2}}(X_1(t), X_2(t), \Omega_{X_1,X_2}(t)) = X_1(t)X_2(t)(1 - \Omega_{X_1,X_2}(t)) + \boldsymbol{\mu}\Omega_{X_1,X_2}(t)\quad (7)$$

Therefore the combination function for Ω_{X_1,X_2} in the description in Fig. 2 is:

$$\mathbf{c}_{\Omega_{X_1,X_2}}(V_1(t), V_2(t), W) = V_1V_2(1 - W) + \boldsymbol{\mu}W = V_1V_2 - V_1V_2W + \boldsymbol{\mu}W\quad (8)$$

Next consider state X_2 . Suppose the original situation depicted in Fig. 1 is described by the combination function $\mathbf{c}_{X_2}(V_1, V_2)$ for X_2 which is applied to the impacts $\omega_{X_1,X_2}(t)X_1(t)$ and $\omega_{X_3,X_2}X_3(t)$ from X_1 and X_3 on X_2 to obtain (based on (1) above) the difference equation for X_2

$$X_2(t + \Delta t) = X_2(t) + \boldsymbol{\eta}_{X_2}[\mathbf{c}_{X_2}(\omega_{X_1,X_2}(t)X_1, \omega_{X_3,X_2}X_3(t)) - X_2(t)]\Delta t\quad (9)$$

In the new situation depicted in Fig. 2 the weight ω_{X_1,X_2} is represented by a state Ω_{X_1,X_2} with activation values $\Omega_{X_1,X_2}(t)$ the same as the connection weight values $\omega_{X_1,X_2}(t)$ in the old situation for each t : $\Omega_{X_1,X_2}(t) = \omega_{X_1,X_2}(t)$. Now there are not two but three states with impact on X_2 , namely X_1 , X_3 and Ω_{X_1,X_2} . This requires a new combination function $\mathbf{c}_{X_2}^*(V_1, V_2, W)$ for X_2 with three arguments, which is applied to the impacts $X_1(t)$, $\omega_{X_3,X_2}X_3(t)$ and $\Omega_{X_1,X_2}(t)$ on X_2 , obtaining $\mathbf{c}_{X_2}^*(X_2(t), \omega_{X_3,X_2}X_3(t), \Omega_{X_1,X_2}(t))$ used in the difference equation for X_2

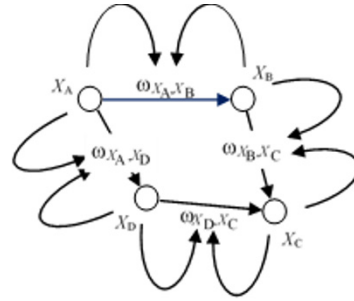
$$X_2(t + \Delta t) = X_2(t) + \boldsymbol{\eta}_{X_2}[\mathbf{c}_{X_2}^*(X_2(t), \omega_{X_3,X_2}X_3(t), \Omega_{X_1,X_2}(t)) - X_2(t)]\Delta t\quad (10)$$

This impact $\mathbf{c}_{X_2}^*(X_2(t), \omega_{X_3,X_2}X_3(t), \Omega_{X_1,X_2}(t))$ is equal to $\mathbf{c}_{X_2}(\omega_{X_1,X_2}(t)X_2(t), \omega_{X_3,X_2}X_3(t))$ in the previous model representation depicted in Fig. 1: $\mathbf{c}_{X_2}^*(X_2(t), \omega_{X_3,X_2}X_3(t), \Omega_{X_1,X_2}(t)) = \mathbf{c}_{X_2}(\omega_{X_1,X_2}(t)X_2(t), \omega_{X_3,X_2}X_3(t))$. So, recalling that $\Omega_{X_1,X_2}(t) = \omega_{X_1,X_2}(t)$ for all t , the new combination function can be defined as $\mathbf{c}_{X_2}^*(V_1, V_2, W) = \mathbf{c}_{X_2}(WV_1, V_2)$. For example, if $\mathbf{c}_{X_2}(V_1, V_2)$ is the sum function $V_1 + V_2$, then $\mathbf{c}_{X_2}^*(V_1, V_2, W) = WV_1 + V_2$ which is a combination of a product and a sum function.

4 Modelling Evolving Social Interactions by Adaptive Networks

Next an adaptive temporal-causal network model is discussed to model evolving social interactions based on the homophily principle. According to this principle, also indicated as birds of a feather flock together, connections are strengthened if the connected states are similar. For example, when two persons both like the same type of music, movies, drinks, and parties, they may strengthen their connection. For the current model the dynamic connection weights ω_{X_A, X_B} from state X_A of person A to state X_B of person B are assumed to change over time based on the principle that the closer the activation levels of the states of the interacting persons, the stronger the mutual connections between the persons will become, and the higher the difference between the activation levels, the weaker they will become. For a conceptual representation, see Fig. 3. Similar to the case of Hebbian learning in Section 3,

Fig. 3: Graphical conceptual representation of an adaptive temporal-causal network model for the homophily principle.



ω_{X_A, X_B} is represented by state Ω_{X_A, X_B} and the weights of the connections involving Ω_{X_A, X_B} are assumed 1: the weights of the connections from X_A and X_B to Ω_{X_A, X_B} , and from Ω_{X_A, X_B} to X_B and to itself. According to the temporal-causal network approach, the homophily principle may be formalised using the following general format of equations (1) and (2) above and a combination function $\mathbf{c}_{A,B}(V_1, V_2, W)$ that still has to be determined:

$$\Omega_{X_A, X_B}(t + \Delta t) = \Omega_{X_A, X_B}(t) + \eta_{\Omega_{X_A, X_B}} [\mathbf{c}_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) - \Omega_{X_A, X_B}(t)] \Delta t \quad (11)$$

$$d\Omega_{X_A, X_B}(t)/dt = \eta_{\Omega_{X_A, X_B}} [\mathbf{c}_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) - \Omega_{X_A, X_B}(t)] \quad (12)$$

Note that the connection weight Ω_{X_A, X_B} increases when $\mathbf{c}_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) > \Omega_{X_A, X_B}(t)$, decreases when $\mathbf{c}_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) < \Omega_{X_A, X_B}(t)$ and stays the same when $\mathbf{c}_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) = \Omega_{X_A, X_B}(t)$.

Examples of such combination functions can be obtained when a threshold value $\tau_{\Omega_{X_A, X_B}}$ is assumed such that the connection weight Ω_{X_A, X_B} becomes stronger when $|X_A(t) - X_B(t)| < \tau_{\Omega_{X_A, X_B}}$ (levels of X_A and X_B close to each other) and weaker when $|X_A(t) - X_B(t)| > \tau_{\Omega_{X_A, X_B}}$ (levels of X_A and X_B not so close to each other). The following is an example which is linear in $X_A(t)$ and $X_B(t)$:

$$\mathbf{c}_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) = \Omega_{X_A, X_B}(t) + \gamma(\tau_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)|) \quad (13)$$

The factor α can be made dependent on $\Omega_{X_A, X_B}(t)$, to keep values of $\Omega_{X_A, X_B}(t)$ within the $[0, 1]$ interval: $\alpha = \Omega_{X_A, X_B}(t)(1 - \Omega_{X_A, X_B}(t))$. This makes the combination function

$$\mathbf{c}_{\Omega_{X_A, X_B}}(V_1, V_2, W) = W + W(1 - W)(\boldsymbol{\tau}_{\Omega_{X_A, X_B}} - |V_1 - V_2|) \quad (14)$$

where V_1, V_2 refer to X_A, X_B and W to Ω_{X_A, X_B} . Thus the following is obtained:

$$\Omega_{X_A, X_B}(t + \Delta t) = \Omega_{X_A, X_B}(t) + \boldsymbol{\eta}_{\Omega_{X_A, X_B}} [\Omega_{X_A, X_B}(t)(1 - \Omega_{X_A, X_B}(t))(\boldsymbol{\tau}_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)|)] \Delta t \quad (15)$$

$$d\Omega_{X_A, X_B}(t)/dt = \boldsymbol{\eta}_{\Omega_{X_A, X_B}} [\Omega_{X_A, X_B}(t)(1 - \Omega_{X_A, X_B}(t))(\boldsymbol{\tau}_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)|)] \quad (16)$$

The combination function for X_B can be found in the same way as in Section 3 for X_2 .

5 Mathematical Analysis of Temporal-Causal Network Models

In this section it is discussed how some types of dynamic properties of adaptive temporal-causal network models can be analysed mathematically, in particular, stationary points and monotonicity. A stationary point of a state occurs at some point in time if for this time point no change occurs: the graph is horizontal at that point. Stationary points are usually maxima or minima (peaks or dips) but sometimes also other stationary points may occur. An equilibrium occurs when for all states no change occurs. From the difference or differential equations describing the dynamics for a model it can be analysed when stationary points or equilibria occur. Moreover, it can be found when a certain state is increasing or decreasing, when a state is not in a stationary point or equilibrium. First a definition for these notions.

Definition (stationary point, increase, decrease, and equilibrium)

- a state Y has a *stationary point* at t if $dY(t)/dt = 0$
- a state Y is *increasing* at t if $dY(t)/dt > 0$
- a state Y is *decreasing* at t if $dY(t)/dt < 0$

The model is in *equilibrium* at t if every state Y of the model has a stationary point at t . This equilibrium is *attracting* when for any state Y , all values of Y in some neighbourhood of the equilibrium value increase when the value is below the equilibrium value and decrease when the value is above the equilibrium value.

A question that can be addressed is whether observations based on one or more simulation experiments are in agreement with a mathematical analysis. If it is found out that the observations are in agreement with the mathematical analysis, then this provides some extent of evidence that the implemented model is correct. If they turn out not to be in agreement with the mathematical analysis, then this indicates that probably there is something wrong, and further inspection and correction has to be initiated. Considering the differential equation (2) for a temporal-causal network model, more specific criteria can be found:

$$dY(t)/dt = \boldsymbol{\eta}_Y [\mathbf{c}_Y(\boldsymbol{\omega}_{X_1, Y} X_1(t), \dots, \boldsymbol{\omega}_{X_k, Y} X_k(t)) - Y(t)] \quad (17)$$

where X_1, \dots, X_k are the states with connections to Y . For example, it can be concluded that

$$dY(t)/dt > 0 \Leftrightarrow c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) > Y(t) \quad (18)$$

In this manner the following criteria can be found.

Criteria for increase, decrease, stationary point and equilibrium

Let Y be a state and X_1, \dots, X_k the states connected toward Y . Then the following hold

$$\begin{aligned} Y \text{ has a stationary point at } t &\Leftrightarrow c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) = Y(t) \\ Y \text{ is increasing at } t &\Leftrightarrow c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) > Y(t) \\ Y \text{ is decreasing at } t &\Leftrightarrow c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) < Y(t) \\ \text{The model is in equilibrium at } t &\Leftrightarrow c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) = Y(t) \\ &\text{for every state } Y \end{aligned}$$

Note that these criteria can immediately be found from a conceptual representation of a temporal-causal network model, as long as the referred combination function is known. Using the above criteria no further numerical representation is needed of the difference or differential equations, for example. From these criteria more insight can be obtained about the behavior of the network model, in particular which stationary points are possible for a state in the model, and which equilibria are possible for the whole model. Sometimes the stationary point equation can be rewritten into an equation of the form $Y(t) = \dots$ such that $Y(t)$ does not occur in the right hand side. In Sections 6 and 7 examples of this are shown.

The criteria can also be used to verify (the implementation of) the model based on inspection of stationary points or equilibria, in two different manners A. and B. Note that in a given simulation the stationary points that are identified are usually approximately stationary; how closely they are approximated depends on different aspects, for example on the step size, or on how long the simulation is done.

A. Verification by checking stationary points through substitution of the values from a simulation in the criterion

1. Generate a simulation
2. Consider any state Y with a stationary point at any time point t and states X_1, \dots, X_k affecting it
3. Substitute the values $Y(t)$ and $X_1(t), \dots, X_k(t)$ in the criterion $c_Y(\omega_{X_1,Y}X_1(t), \dots, \omega_{X_k,Y}X_k(t)) = Y(t)$
4. If the equation holds (for example, with an accuracy $< 10^2$), then this test succeeds, otherwise it fails
5. If this test fails, then it has to be explored were the error can be found

Note that this method A. works without having to solve the equations, only substitution takes place; therefore it works for any choice of combination function. Moreover, note that the method also works when the values of the states fluctuate, for example according to a recurring pattern (a limit cycle). In such cases for each state there

are maxima (peaks) and minima (dips) which also are stationary points to which the method can be applied; here it is important to choose a small step size as each stationary point occurs at one time point only. There is still another method B. possible that can be applied sometimes; it is based on solving the equations for the stationary point values by symbolic rewriting. This can provide explicit expressions for stationary point values in terms of the parameters of the model. Such expressions can be used to predict equilibrium values for specific simulations, based on the choice of parameter values. For more details, see [26], Chapter 12, or [28]. This method B. provides more, but a major drawback is that it cannot be applied in all situations; this depends on the chosen combination functions; e.g., for logistic functions it does not work.

6 Mathematical Analysis for Hebbian Learning

It can be analysed from the network model from Section 3 when a Hebbian adaptation process has a stationary point and when it increases or decreases. Recall equation (8):

$$c_{\Omega_{X_1, X_2}}(V_1, V_2, W) = V_1 V_2 (1 - W) + \mu W \quad (19)$$

where V_1 , V_2 refer to $X_1(t)$, $X_2(t)$ and W to $\Omega_{X_1, X_2}(t)$. According to the criteria in Section 5 a stationary point of $\Omega_{X_1, X_2}(t)$ occurs if and only if:

$$c_{\Omega_{X_1, X_2}}(X_1(t), X_2(t), \Omega_{X_1, X_2}(t)) = \Omega_{X_1, X_2}(t) \quad (20)$$

which for this case is equivalent to the following three rewritten forms

$$\begin{aligned} X_1(t)X_2(t)(1 - \Omega_{X_1, X_2}(t)) + \mu \Omega_{X_1, X_2}(t) &= \Omega_{X_1, X_2}(t) \\ X_1(t)X_2(t) - X_1(t)X_2(t)\Omega_{X_1, X_2}(t) - (1 - \mu)\Omega_{X_1, X_2}(t) &= 0 \\ X_1(t)X_2(t) &= (X_1(t)X_2(t) + (1 - \mu))\Omega_{X_1, X_2}(t) \end{aligned} \quad (21)$$

Note that for $\mu = 1$ (fully persistent) this reduces to

$$X_1(t)X_2(t) = X_1(t)X_2(t)\Omega_{X_1, X_2}(t) \quad (22)$$

and for $\mu < 1$ it can be rewritten into

$$\Omega_{X_1, X_2}(t) = \frac{X_1(t)X_2(t)}{1 - \mu + X_1(t)X_2(t)} \quad (23)$$

Thus two cases are found:

Stationary points for $\Omega_{X_1, X_2}(t)$ when $\mu = 1$ (fully persistent, no extinction)

When $\mu = 1$ a stationary point occurs for Ω_{X_1, X_2} if and only if

$$\begin{aligned} X_1(t) = 0 \text{ or } X_2(t) = 0 \text{ and } \Omega_{X_1, X_2}(t) \text{ has any value} \\ \text{or } \Omega_{X_1, X_2}(t) = 1 \text{ and } X_1(t) \text{ and } X_2(t) \text{ have any values} \end{aligned}$$

Stationary points for $\Omega_{X_1, X_2}(t)$ when $\mu < 1$ (not fully persistent, some extinction)

For $\mu < 1$ a stationary point occurs for Ω_{X_1, X_2} if and only if $\Omega_{X_1, X_2} = \frac{X_1(t)X_2(t)}{1 - \mu + X_1(t)X_2(t)}$.

In particular for $\mu < 1$ a stationary point occurs if and only if

- (a) $\Omega_{X_1, X_2}(t) = \frac{1}{1 + (1 - \mu)/(X_1(t)X_2(t))}$ and both $X_1(t) > 0$ and $X_2(t) > 0$
- (b) $\Omega_{X_1, X_2}(t) = 0$ and $X_1(t) = 0$ or $X_2(t) = 0$

Note that the above conditions show that when both $X_1(t) > 0$ and $X_2(t) > 0$, a positive stationary point value is found, which is 1 for $\mu = 1$, and $\frac{1}{1+(1-\mu)/(X_1(t)X_2(t))}$ for $\mu < 1$ which is nonzero and < 1 . So without extinction the value 1 is possible, but extinction always makes it < 1 . In fact the maximal value of this occurs when both $X_1(t) = 1$ and $X_2(t) = 1$, in which case the stationary point value is $\frac{1}{2-\mu}$. It turns out that this is the maximal value a stationary point can have, and this value is < 1 when $\mu < 1$. For example, for $\mu = 0.95$, and $X_1(t) = 1$ and $X_2(t) = 1$, the positive stationary point value for $\Omega_{X_1, X_2}(t)$ is about 0.95. Another example is $\mu = 0.8$, and $X_1(t) = 1$ and $X_2(t) = 1$, in which case the stationary point value is 0.83. In further analysis of the criteria for increase and decrease it turns out that for given (positive) values of $X_1(t)$ and $X_2(t)$ the value of $\Omega_{X_1, X_2}(t)$ increases when it is under the positive stationary point value and it decreases when it is above this value (the value is attracting):

Increasing Ω_{X_1, X_2} when $X_1(t) > 0$ and $X_2(t) > 0$:

$$d\Omega_{X_1, X_2}(t)/dt > 0 \Leftrightarrow \Omega_{X_1, X_2}(t) < \frac{1}{1+(1-\mu)/(X_1(t)X_2(t))}$$

Decreasing Ω_{X_1, X_2} when $X_1(t) > 0$ and $X_2(t) > 0$:

$$d\Omega_{X_1, X_2}(t)/dt < 0 \Leftrightarrow \Omega_{X_1, X_2}(t) > \frac{1}{1+(1-\mu)/(X_1(t)X_2(t))}$$

For comparison to example simulation patterns showing the behaviours analysed above, see [26], Chapter 12.

7 Mathematical Analysis for the Homophily Principle

In Section 4 it was shown how the homophily principle for evolving social interaction may be modeled using a combination function (see equation (14))

$$c_{\Omega_{X_A, X_B}}(V_1, V_2, W) = W + W(1 - W)(\tau_{\Omega_{X_A, X_B}} - |V_1 - V_2|) \quad (24)$$

In this section it is analysed which stationary points can occur for $\Omega_{X_A, X_B}(t)$, according to the approach described in Section 5. For this case the criterion from Section 5 for a stationary point is:

$$\begin{aligned} c_{\Omega_{X_A, X_B}}(X_A(t), X_B(t), \Omega_{X_A, X_B}(t)) = \Omega_{X_A, X_B}(t) \Leftrightarrow \\ \Omega_{X_A, X_B}(t)(1 - \Omega_{X_A, X_B}(t))(\tau_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)|) = 0 \end{aligned} \quad (25)$$

Clearly for $\Omega_{X_A, X_B}(t) = 0$ or $\Omega_{X_A, X_B}(t) = 1$ one of the left hand side factors in this condition is 0. In contrast, when $0 < \Omega_{X_A, X_B}(t) < 1$ the right hand factor should be 0:

$$\tau_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)| = 0 \Leftrightarrow |X_A(t) - X_B(t)| = \tau_{\Omega_{X_A, X_B}} \quad (26)$$

So, in principle there are three types of stationary points for $\Omega_{X_A, X_B}(t)$.

Stationary points for $\Omega_{X_A, X_B}(t)$

$\Omega_{X_A, X_B}(t) = 0$ or $\Omega_{X_A, X_B}(t) = 1$ or $|X_A(t) - X_B(t)| = \tau_{\Omega_{X_A, X_B}}$ and $\Omega_{X_A, X_B}(t)$ has any value

Similarly the following can be found.

Increasing $\Omega_{X_A, X_B}(t)$

$$d\Omega_{X_A, X_B}(t)/dt > 0 \Leftrightarrow (\tau_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)|) > 0 \Leftrightarrow |X_A(t) - X_B(t)| < \tau_{\Omega_{X_A, X_B}}$$

Decreasing $\Omega_{X_A, X_B}(t)$

$$d\Omega_{X_A, X_B}(t)/dt < 0 \Leftrightarrow (\tau_{\Omega_{X_A, X_B}} - |X_A(t) - X_B(t)|) < 0 \Leftrightarrow |X_A(t) - X_B(t)| > \tau_{\Omega_{X_A, X_B}}$$

This shows that for cases that $|X_A(t) - X_B(t)| < \tau_{\Omega_{X_A, X_B}}$ the connection keeps on becoming stronger until $\Omega_{X_A, X_B}(t)$ approaches 1. Similarly for cases that $|X_A(t) - X_B(t)| > \tau_{\Omega_{X_A, X_B}}$ the connection keeps on becoming weaker until $\Omega_{X_A, X_B}(t)$ approaches 0. This implies that $\Omega_{X_A, X_B}(t) = 0$ and $\Omega_{X_A, X_B}(t) = 1$ can both become attracting, but under different circumstances concerning the values of $X_A(t)$ and $X_B(t)$. In [26], Chapter 11, Section 11.7 for such an adaptive network model an example simulation is shown where indeed the connection weights all converge to 0 or 1, and during this process clusters are formed of persons with equal levels of their state; see also [24].

8 Discussion

The Network-Oriented Modelling approach based on adaptive temporal-causal networks as described here (see also [25, 26]), provides a dynamic modelling approach that enables a modeller to design high level conceptual model representations in the form of cyclic graphs (or connection matrices). These conceptual representations can be systematically transformed in an automated manner into executable numerical representations that can be used to perform simulation experiments. The modelling approach makes it easy to take into account on the one hand theories and findings from any domain from, for example, biological, psychological, neurological or social sciences, as such theories and findings are often formulated in terms of causal relations. This applies, among others, to mental processes based on complex brain networks, which, for example, often involve dynamics based on interrelating and adaptive cycles, but equally well it applies to the adaptive dynamics of social interactions. This enables to address complex adaptive phenomena within all kinds of integrated cognitive, affective and social processes. By using temporal-causal relations from those domains as a main vehicle and structure for network models, the obtained network models get a strong relation to the large body of empirically founded knowledge from the Neurosciences and Social Sciences. This makes them scientifically justifiable to an extent that is not attainable for black box models which lack such a relation.

In this paper it was discussed in some detail how mathematical analysis can be used to find out some properties of the dynamics of a network model designed according to a Network-Oriented Modelling approach based on temporal-causal networks; see also [26], Chapter 12, or [28]. An advantage is that such an analysis is done without performing simulations. This advantage makes that it can be used as an additional source of knowledge, independent of a specific implementation of the model. By comparing properties found by mathematical analysis and properties observed in simulation experiments a form of verification can be done. If a discrepancy is found, for example, in the sense that the mathematical analysis predicts a certain property but some simulation does not satisfy this property, this can be a reason to inspect

the implementation of the model carefully (and/or check whether the mathematical analysis is correct). Having such an option can be fruitful during a development process of a model, as to acquire empirical data for validation of a model may be more difficult or may take a longer time.

References

- [1] Beer, R.D.: On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior* **3**(4), 469–509 (1995)
- [2] Brauer, F., Nohel, J.A.: *The qualitative theory of ordinary differential equations: an introduction*. Courier Corporation (2012)
- [3] Chung, B., Choi, H., Kim, S.: Workflow-enabled internet service delivery for a variety of access networks. In: *APNOMS'03* (2003)
- [4] Cottret, L., Jourdan, F.: Graph methods for the investigation of metabolic networks in parasitology. *Parasitology* **137**(9), 1393–1407 (2010)
- [5] Dubois, D., Lang, J., Prade, H.: Fuzzy sets in approximate reasoning, part 2: logical approaches. *Fuzzy sets and systems* **40**(1), 203–244 (1991)
- [6] Dubois, D., Prade, H.: Possibility theory, probability theory and multiple-valued logics: A clarification. *Annals of mathematics and Artificial Intelligence* **32**(1–4), 35–66 (2001)
- [7] Elzas, M.S.: Organizational structures for facilitating process innovation. In: *Real Time Control of Large Scale Systems*, pp. 151–163. Springer (1985)
- [8] Felsen, L.B., Mongiardo, M., Russer, P.: Electromagnetic field representations and computations in complex structures i: Complexity architecture and generalized network formulation. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields* **15**(1), 93–107 (2002)
- [9] Felsen, L.B., Mongiardo, M., Russer, P.: *Electromagnetic field computation by network methods*. Springer Science & Business Media (2009)
- [10] Gerla, G.: *Fuzzy logic: mathematical tools for approximate reasoning*, vol. 11. Springer Science & Business Media (2013)
- [11] Gerstner, W., Kistler, W.M.: Mathematical formulations of hebbian learning. *Biological cybernetics* **87**(5–6), 404–415 (2002)
- [12] Grossberg, S.: On learning and energy-entropy dependence in recurrent and nonrecurrent signed networks. *Journal of Statistical Physics* **1**(2), 319–350 (1969)
- [13] Hebb, D.O.: *The organization of behavior* (1949)
- [14] Hirsch, M.W.: Convergent activation dynamics in continuous time networks. *Neural Networks* **2**(5), 331–349 (1989)
- [15] Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences* **79**(8), 2554–2558 (1982)
- [16] Hopfield, J.J.: *Neurocomputing: Foundations of research*. chap. Neurons with Graded Response Have Collective Computational Properties Like Those of Two-state Neurons, pp. 577–583. MIT Press, Cambridge, MA, USA (1988). URL <http://dl.acm.org/citation.cfm?id=656669.104438>
- [17] Laurent, H.: *Trait d'analyse*, vol.1. Gauthier-Villars, Paris (1891). URL <https://cds.cern.ch/record/460639>
- [18] Laurent, H.: *Trait d'analyse*, vol.2. Gauthier-Villars, Paris (1893). URL <https://cds.cern.ch/record/460639>
- [19] Lotka, A.: *Elements of Physical Biology*. Williams & Wilkins Company (1925). URL <https://books.google.it/books?id=lsPQAAAAAMAAJ>
- [20] Naud, A., Le Maitre, D., de Jong, T., Mans, G.F.G., Hugo, W.: (2008)
- [21] Poincaré, H.: *New methods of celestial mechanics*, vol. 13. Springer Science & Business Media (1992)

- [22] Poincar, H.: Mmoire sur les courbes dfinies par une quation diffrentielle (ii). *Journal de Mathematiques Pures et Appliques* **8**, 251–296 (1882)
- [23] Russer, P., Cangellaris, A.C.: Network-oriented modeling, complexity reduction and system identification techniques for electromagnetic systems. In: *Proc. 4th Int. Workshop on Computational Electromagnetics in the Time-Domain: TLM/FDTD and Related Techniques*, pp. 105–122 (2001)
- [24] Sharpanskykh, A., Treur, J.: Modelling and analysis of social contagion in dynamic networks. *Neurocomputing* **146**, 140–150 (2014)
- [25] Treur, J.: Dynamic modeling based on a temporal-causal network modeling approach. *Biologically Inspired Cognitive Architectures* **16**, 131–168 (2016)
- [26] Treur, J.: *Network-Oriented Modeling: Addressing Complexity of Cognitive, Affective and Social Interactions. Understanding Complex Systems*. Springer International Publishing (2016). URL <https://books.google.it/books?id=LcowDQAAQBAJ>
- [27] Treur, J.: *Network-Oriented Modeling and Its Conceptual Foundations*, pp. 3–33. Springer International Publishing, Cham (2016)
- [28] Treur, J.: Verification of temporal-causal network models by mathematical analysis. *Vietnam Journal of Computer Science* **3**(4), 207–221 (2016)
- [29] Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems* **100**, 9–34 (1999)

What governs a language's lexicon? Determining the organizing principles of phonological neighbourhood networks

Rory Turnbull and Sharon Peperkamp

Abstract The lexicons of natural language can be characterized as a network of words, where each word is linked to phonologically similar words. These networks are called phonological neighbourhood networks (PNNs). In this paper, we investigate the extent to which observed properties of these networks are mathematical consequences of the definition of PNNs, consequences of linguistic restrictions on what possible words can sound like (phonotactics), or consequences of deeper cognitive constraints that govern lexical development. To test this question, we generate random lexicons, with a variety of methods, and derive PNNs from these lexicons. These PNNs are then compared to a real network. We conclude that most observed characteristics of PNNs are either intrinsic to the definition of PNNs, or are phonotactic effects. However, there are some properties—such as extreme assortativity by degree—which may reflect true cognitive organizing principles.

1 Introduction

In natural languages, sentences are composed of words, which are in turn composed of strings of symbols referred to as *phonemes*, which represent the smallest units of sound that can be used to distinguish words from each other. Many psycholinguistic theories of spoken word recognition and infant language acquisition rely on a concept of the phonological similarity of words, termed *neighbourhood*, which is defined in terms of the phonemic structure of words. Two words are neighbours of each other if they differ by the deletion, addition, or substitution of one and only one segment—that is, an edit distance of one. For example, neighbours of *plan* include

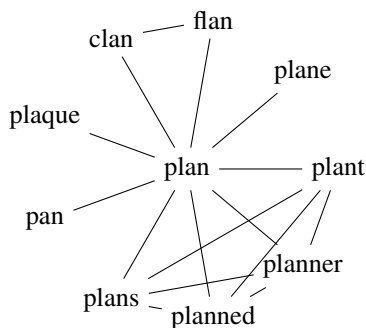
Rory Turnbull (e-mail: rory.turnbull@ens.fr) · Sharon Peperkamp (e-mail: sharon.peperkamp@ens.fr)

Laboratoire de Sciences Cognitives et Psycholinguistique (ENS, EHESS, CNRS)

Rory Turnbull · Sharon Peperkamp

Département d'Etudes Cognitives, Ecole Normale Supérieure, PSL Research University, 29 rue d'Ulm, 75005 Paris, France

Fig. 1 Example phonological neighbourhood network centred around the English word *plan*. Note that it is the sound of a word, not the spelling, which determines the phonological neighbours. Note further that some neighbours of a word are neighbours of each other.



pan (deletion of /l/), *plant* (addition of /t/), and *clan* (substitution of /k/ for /p/). See Figure 1 for a visual example.¹ The neighbourhood relation is symmetric, intransitive and anti-reflexive.

For a given lexicon, then, it is possible to construct a complex network to model phonological neighbourhood relations throughout the language. Phonological neighbourhood networks (PNNs) have been used to study aspects of lexical organization in several languages [1, 16, 20]. In this paper, we explore the extent to which these complex network analyses can provide insight into the psychological organization of human language.

Vitevitch [20] first proposed the use of PNNs to study the phonological aspects of lexicons. In a PNN, every word in the lexicon is a vertex in a graph, and two vertices are linked by an edge if a neighbourhood relation obtains between the two words. This process yields an undirected, unweighted graph, ideal for examination with the tools of complex network analysis.

Early work on PNNs, in a variety of languages, has demonstrated that these networks have distinct properties which differ in important ways from other complex networks studied in the literature [1, 20]. For example, while most complex networks typically have a giant component which contains around 80–90% of the vertices, the observed values for PNNs fall between 10% and 65% [1, 16]. PNNs were also found to be remarkably robust to vertex removal, with the average shortest path length remaining the same when up to 5% of vertices were removed. Notably, this effect held regardless of whether vertex removal was at random or in order of degree [1]. Despite these differences from other networks, the high clustering coefficients established that PNNs exhibit small world properties.

However, these statistics and examinations rely on comparing the observed networks to random networks [14]. While this approach is reasonable for many kinds of complex networks, it is not an appropriate comparison for PNNs. Unlike other networks, where vertices exist independently of each other and edges can be made

¹ Note that neighbourhood is defined based on the pronunciation of a word, not the spelling. For instance, while the spelling of the words *knee* and *neat* are quite different, the pronunciations are very similar. The addition or deletion of the /t/ sound will transform *knee* into *neat* and vice versa. Therefore, these words are neighbours. On the other hand, the words *tough* and *though* have very similar spellings, but their pronunciations—/tʌf/ and /ðʊ/ respectively—are very different. These words are not neighbours.

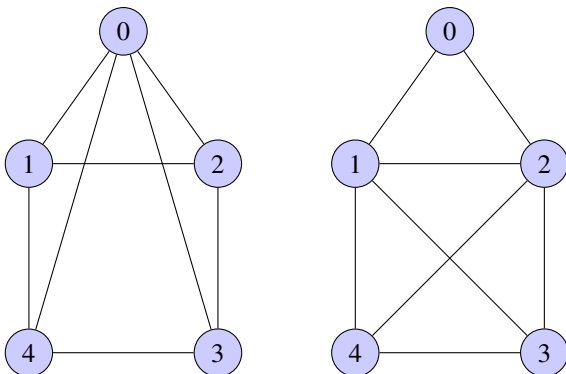
or unmade (for example, friendships made or broken, shipping routes established or abandoned), in a PNN the edges (neighbourhood relations) are intrinsic to the definition of the vertices themselves (the phonological structure of the words). That is, because edges exist between two vertices if and only if the two words are phonological neighbours, there are certain graphs which are not possible PNNs.

One such graph is shown in the left of Fig. 2. Here, each vertex is connected to every other vertex, with two exceptions: vertices 1 and 3 are not connected, and vertices 2 and 4 are not connected. It is not possible for this graph to have its vertices labelled such that the shortest path from vertex-to-vertex is equal to the edit distance (Hamming distance) of the vertex labels [9]. In other words, this graph cannot represent neighbourhood relations between words. On the other hand, the graph on the right of Fig. 2 is plausibly a PNN, with the mapping $0=cant, 1=can, 2=cat, 3=cab, 4=cap$.² Note that the graphs in Fig. 2 both have the same number of vertices and edges, but the left one could not be a PNN while the right one could be.

The difference between these graphs is that the graph on the right is *addressable*, that is, there exists a vertex labelling schema which satisfies the neighbourhood relation, while the graph on the left is non-addressable [2]. Addressable graphs have also been termed ℓ_1 -graphs, as it can be shown that addressable graphs are isometrically embeddable into a hypercube [6, 17]. Since the distances along the edges of a hypercube fall under the definition of an ℓ_1 metric, it follows that these graphs are isometrically embeddable into an ℓ_1 metric space [7]. The recognition of such graphs can be solved in polynomial time [8, 11].

For these reasons, random graphs are inappropriate as comparison cases when considering PNNs. Currently, it is not easy to tell if results obtained are generalizable results about language and lexical organization, or if they are simply consequences of the structure of an addressable graph [10]. It has further been noted that the statistics of PNNs are very sensitive to the distribution of word lengths within a lexicon and the number of phonemes in the language’s symbol set [16, 19]. For example, given n phonemes, the number of possible words is an exponential term of

Fig. 2 Two graphs, both with the same number of vertices and edges. The graph on the left is non-addressable. This graph could not represent a PNN. The graph on the right is addressable. This graph could be represent a PNN. Consider the mapping $0=cant, 1=can, 2=cat, 3=cab, 4=cap$. Each vertex is connected to its phonological neighbours.



² Other possible mappings include $(0=slow, 1=low, 2=sew, 3=go, 4=show)$; $(0=lamp, 1=lamb, 2=lap, 3=lab, 4=lad)$; $(0=gasp, 1=gas, 2=gap, 3=gag, 4=gash)$; $(0=iode, 1=eyed, 2=ode, 3=aid, 4=add)$ and so on.

n , while the number of possible neighbourhood connections is a linear term of n [19]. This fact has consequences for how cross-linguistic comparisons are carried out, as languages differ in the sizes of their lexicons and their number of phonemes [16]. These difficulties make the use of complex network analysis in the study of PNNs a complex undertaking.

In this study, we generate random lexicons, rather than random graphs. PNNs are derived from these random lexicons, guaranteeing that the resulting graphs are addressable. These simulated PNNs can be compared to real PNNs. In broad terms, there are two possible outcomes to this investigation:

1. The simulated PNNs are indistinguishable from a real PNN.
2. The simulated PNNs differ from a real PNN.

In the case of (1), we can conclude that alleged properties of the human language faculty relating to lexical organization [1] are simply consequences of the mathematical structure of PNNs. In this regard, the results could shed light on the hypercube-embeddable graphs, but not on language.

In the case of (2), we can conclude that any areas of difference between the simulated PNN and the real PNN are due to some organizing principle or cognitive constraint operating on language. For example, to ensure efficient communication, the lexicon may be organized to avoid having words which sound very similar [12].

2 Method

To address the question of which properties of PNNs are simply due to their definition and which are due to linguistic principles, we generated random lexicons, derived PNNs from these lexicons, and compared the properties of these PNNs to the PNN of English. The PNN of English we used was derived from the Hoosier Mental Lexicon [15], a dictionary of American English with phonological transcriptions of 19,320 words, after homophone removal. We refer to this lexicon and PNN as the ‘real English lexicon’ and ‘real English PNN’ to distinguish it from the simulated (random) lexicons and PNNs that we generated.

2.1 *Random lexicons*

Each random lexicon had the same size and mean word length (6.35 phonemes), and used the same inventory of phonemes, as the real English lexicon. Five groups of random lexicons were generated, differing in the extent to which they approximate the real English lexicon: uniform random lexicons; Zipfian random lexicons; scrambled random lexicons; bigram random lexicons; and trigram random lexicons. Each group consisted of 200 random lexicons.

The simplest group was the **uniform random lexicons**, which were created by randomly sampling from the phoneme inventory in a uniform manner. Word length was sampled from a Poisson distribution (with $\lambda = 6.35$). In these lexicons, while

the overall properties of the lexicon (number and length of words) was the same as that of the real English lexicon, the content of the words resemble what one would obtain from random typing.

Zipfian random lexicons were created in the same manner, except that the sampling from the phoneme inventory was not uniform. Instead, phonemes were frequency ranked according to a Zipf distribution. That is, given N phonemes, the probability of phoneme ϕ_k , where $k \in \{1, \dots, N\}$ is given as

$$p(\phi_k) = \frac{k^{-1}}{\sum_{n=1}^N n^{-1}}.$$

Phoneme distributions in natural languages are approximately Zipfian [21]; these lexicons therefore approximate more closely the structure of English than the uniform random lexicons.

The **scrambled random lexicons** began with the real English lexicon and scrambled the order of the phonemes within each word. This scrambling disrupts the neighbourhood structure of the words, while preserving the overall phoneme frequencies exactly.

Of these three groups, the uniform random group approximates the average word length of English; the Zipfian group the average word length and average phoneme frequency; and the scrambled group matches word length and phoneme frequencies exactly. An important difference between these groups and the real English lexicon is that of *phonotactics*—higher-level generalizations about the combinatoric possibilities of phonemes. The classical example is that neither *blick* nor *bnick* are actual English words, but the former could be a word, while the latter could not. This is due to a restriction in what consonant clusters English permits at the beginning of syllables.³

Due to the lack of phonotactics in the randomly generated lexicons, any differences between them and the real English lexicon could either be due to organizing principles of lexical storage, or simply a consequence of the fact that phonotactics restrict the possible words that can appear in a lexicon. To test for this possibility, the bigram and trigram random lexicon groups were generated.

These random lexicons were generated by creating n -gram models of English phoneme distributions, where $n = 2$ for the **bigram random lexicons** and $n = 3$ for the **trigram random lexicons**. In these models, the probability of a given phoneme is conditioned on the probability of the preceding $n - 1$ phonemes. (Kneser-Ney discounting was applied to smooth the probability space for unobserved forms.) In this way, the model is able to account for basic distributional facts of English phonotactics—for example, vowels and consonants tend to alternate; the consonant cluster ‘thl’ (as in *decathlon*) is rare, but the consonant cluster ‘str’ (as in *string*) is common; and so on. Using this model, a lexicon the same size as the real English lexicon was generated. Due to the fact that the n -gram models encodes the probability of individual phonemes, and the ‘end-of-word’ character, these generated lexicons

³ Note that in some languages, like Russian, both *blick* and *bnick* are possible words, while in others, like Japanese, neither are possible words.

approximate the real English lexicon in terms of phoneme frequencies and mean word length.

The bigram model yields English-like words, but there are exceptions, for example, /#nd/, where # represents the beginning of a word. There are no English words that begin with /nd/.⁴ This situation arises due to the fact that the model can only ‘see’ two phonemes at a time. The sequence /#n/ (that is, the beginning of a word, followed by /n/) is a frequent bigram sequence, and so it has relatively high probability; likewise, the sequence /nd/ is frequent and also has a relatively high probability, and so therefore there is a chance that the model will output sequences like /#nd/. The trigram model, on the other hand, is able to see three phonemes at a time, notes that /#nd/ is not attested in the original lexicon, and accordingly assigns this sequence an extremely low probability. Thus, the trigram model is more English-like than the bigram model. Still, phonotactics are considerably more complex than phoneme-level *n*-gram probabilities, and the trigram model still produces words which sound quite un-English-like. The use of complex phonotactic generators to create ‘English-like’ simulated lexicons can help alleviate this problem [12], but such an investigation is beyond the scope of the current study.

To summarize, in terms of fidelity to English linguistic lexical patterns, these random lexicon groups are expected to follow the following hierarchy:

uniform < Zipfian < scrambled < bigram < trigram

Comparison of these random lexicons with each other and with the real English lexicon allows us to determine which observed properties of English are lexically meaningful. If a property is true of all PNNs, it is likely to be a simple consequence of the definition of the neighbourhood relation over lexicons, and does not necessarily reveal anything about language. If a property is true of the real English PNN and the *n*-gram PNNs, but not the other random PNNs, it is likely to be a consequence of the phonotactic patterns of the lexicon—hard limits on what shapes words can take. If a property is true only of the English PNN but not any of the random PNNs, then it is likely to be due to a deeper organizing principle of the lexicon.

2.2 Network measures

For each group of PNNs, several network measures were taken.

- Giant component size: the size, as a ratio of the number of vertices in the entire graph, of the largest connected component.
- Clustering coefficient: the mean clustering coefficient for each vertex in the entire graph.
- Mean number of neighbours: the mean number of neighbours for each vertex in the entire graph.

⁴ Even in borrowed words like *Ndebele*, a short vowel sound is usually inserted before the /n/.

- Assortativity by degree [13]: the correlation coefficient of the degree of a vertex with that of its neighbours, averaged over the entire graph. This measures the extent to which highly-connected words cluster together.
- Shortest path: the average shortest path length for all pairwise comparisons. Vertices which are not connected are ignored, essentially yielding a grand mean of each connected component weighted by the number of vertices in each component.

2.3 Robustness to vertex removal

To evaluate the relative robustness of each PNN, vertex removal was performed. A proportion of vertices were removed, and the average shortest path of the graph was measured. The procedure was then repeated with a larger proportion of vertices. This procedure allows us to examine the change in the robustness of the network as successively more vertices were removed.

Two vertex removal methods were employed: a random method, where vertices were removed at random; and a targeted method, where vertices were removed in decreasing order of degree. That is, the word with the most neighbours was removed first, the word with the second most was removed second, and so on. We tested removal proportions from 0 to 0.05, in 21 equally-spaced steps. Two measures of network robustness were used: giant component size and average shortest path. We follow convention in assuming that larger giant component size and smaller shortest path represent more robust networks.

3 Results

Table 1 summarizes the results for the real English PNN and of the five groups of random PNNs.

3.1 Overall patterns

For giant component size, clustering coefficient, and mean number of neighbours, the statistics obeyed the following hierarchy:

$$\text{uniform} < \text{Zipfian} < \text{scrambled} < \text{bigram} \approx \text{trigram} \approx \text{English}$$

That is, the n -gram PNNs were very similar to the real English PNN, while the other random PNNs had lower values as a function of their projected similarity to English. Nevertheless, while the other random PNNs were not similar to English, their statistics do indicate some small-world properties, as previously reported [1, 20].

The size of the real English PNN giant component is still smaller than most scale-free networks studied in the literature [14]. The fact that the real English PNN regardless has the largest giant component of all the PNNs suggests that the

English lexicon has clusters of highly-connected words [18]. For this to happen, the lexicon must employ a large degree of *re-use* of common elements and sequences of phonemes. It has been theorized that such re-use is beneficial for the developing lexicon in infant and child language acquisition [3], and aids in the processes of speech production and perception in adults [4, 5].

All the PNNs examined are assortative by degree: words with many neighbours tend to cluster together. Assortativity was higher for the n -gram random PNNs than the other random PNNs, and it was highest of all for the English PNN. Taken together, these results suggest that the property of assortativity in general is intrinsic to PNNs, but that it is enhanced by the presence of phonotactics, and enhanced further by unknown lexical organizational constraints.

The real English PNN had neither the longest nor the shortest mean shortest path length. This value does not appear to readily distinguish the real English PNN from the random PNNs, nor does it distinguish the different random PNNs from each other.

Table 1: Summary statistics for the real English PNN and the five groups of random PNNs. Standard deviations included in parentheses. GC: giant component; Clust.: clustering; Sh.: shortest.

	GC size	Clust. coefficient	Mean # neighbours	Assortativity	Sh. path
Uniform	.023 (.002)	.009 (.001)	0.108 (0.010)	.540 (.045)	6.032 (0.334)
Zipfian	.100 (.003)	.034 (.002)	0.628 (0.032)	.240 (.021)	4.835 (0.073)
Scrambled	.167 (.002)	.046 (.001)	0.710 (0.010)	.427 (.019)	7.057 (0.091)
Bigram	.286 (.004)	.106 (.002)	2.604 (0.050)	.459 (.009)	5.242 (0.038)
Trigram	.371 (.005)	.138 (.002)	3.018 (0.055)	.538 (.008)	6.432 (0.068)
English	.320	.117	2.675	.643	6.991

3.2 Vertex removal

The patterns of robustness to vertex removal are shown in Fig. 3 for giant component size, and Fig. 4 for average shortest path length. For all groups of PNNs, random vertex removal does not appear to influence giant component size, while targeted vertex removal leads to a decline in giant component size. However, it can be seen

that the fall is very sharp for the uniform, Zipfian, and scrambled PNNs (rapidly reaching zero), while the slope is much gentler for the bigram, trigram, and real English PNNs.

The same pattern is observed for the shortest path length: no change for random removal, rapid increase for targeted removal for the uniform, Zipfian, and scrambled PNNs, and gentle increase for targeted removal for the bigram, trigram, and real English PNNs. After a point, the shortest path lengths for the uniform, Zipfian, and scrambled PNNs fall; this is a consequence of the rapid fragmenting of the graph into many isolated islands, and does not reflect an increase in robustness. (Note that the falls coincide with the giant component size approaching zero.)

These results demonstrate that, while the real English PNN is remarkably robust to both random and targeted vertex removal [1], the same is true of the bigram and trigram random PNNs. The observed robustness is therefore not necessarily due to an organizing principle of lexical structure, but phonotactic limitations on possible words.

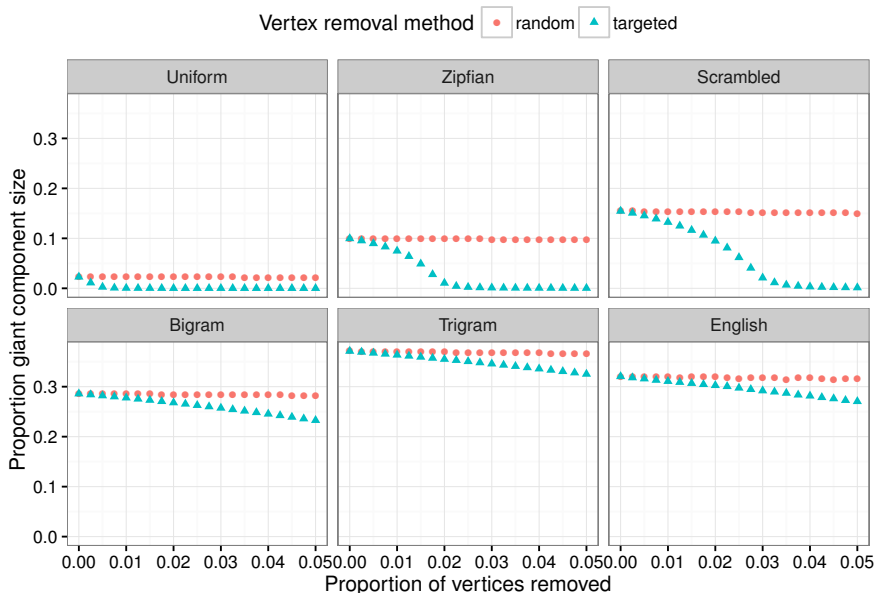


Fig. 3: Giant component size for the five random groups of PNNs, plus the real English PNN, given two vertex removal methods, plotted as a function of the proportion of vertices removed. Red circles depict values for random vertex removal; blue triangles depict values for targeted vertex removal.

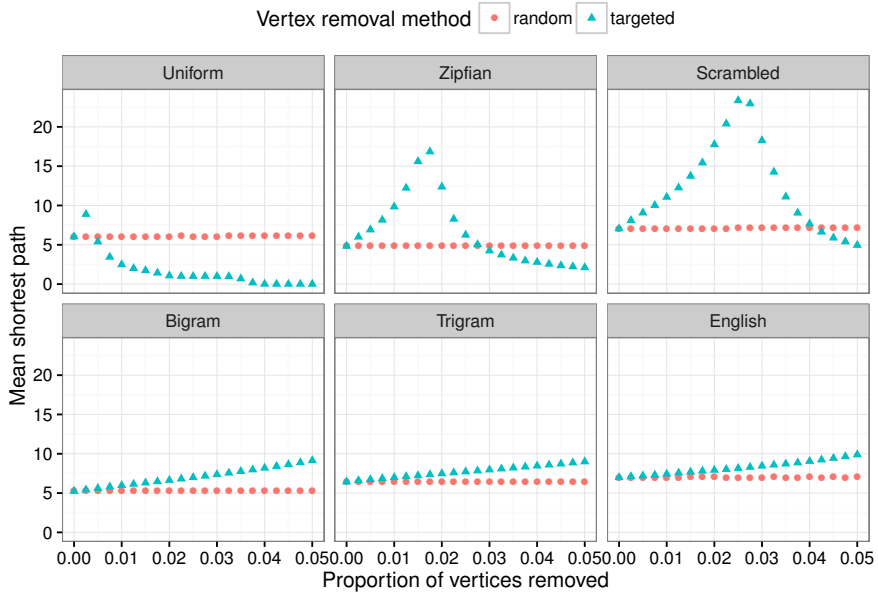


Fig. 4: Shortest path lengths for the five random groups of PNNs, plus the real English PNN, given two vertex removal methods, plotted as a function of the proportion of vertices removed. Red circles depict values for random vertex removal; blue triangles depict values for targeted vertex removal.

4 Discussion

For both the real English PNN and the random PNNs, the clustering coefficients were relatively high, confirming the assertion that PNNs have small-world properties [1]. However, as this was observed for the random PNNs too, it would appear to be a property intrinsic to the definition of a PNN, and therefore not necessarily psycholinguistically meaningful.

In terms of giant component size and mean number of neighbours, the real English PNN was midway between the bigram and trigram random PNNs, suggesting that these properties are due to phonotactics rather than any deeper constraints which may modulate the development of the lexicon.

However, where the real English PNN stood out from the random PNNs was in assortativity by degree. While all the PNNs were assortative, the real English PNN was the most of all. It is possible that this high level of assortativity aids in lexical retrieval by limiting the spread of activation to irrelevant candidate words in the process of speech perception [20]. However, the mechanisms by which the real English PNN obtains this high level of assortativity is unknown.

Finally, the vertex removal analysis demonstrated that while the real English PNN and the n -gram PNNs were very robust to targeted vertex removal, the other random PNNs rapidly lost robustness. In this regard, the non- n -gram random PNNs

are similar to scale-free networks, in that the mean shortest path length rapidly increases upon targeted vertex removal [14]. This finding suggests that the robustness observed by [1] is not necessarily due to a particular cognitive constraint on lexical organization, but a consequence of phonotactics.

5 Conclusion

With a novel method for generation of random PNNs, we have shown that some properties of PNNs—such as small world properties, small giant component size, and assortativity by degree—are due to the definition of the neighbourhood relation that defines PNNs, rather than properties of language *per se*. Others properties are common to the real PNN and n -gram PNNs, which simulate the phonotactic patterns of natural language. For example, the n -gram PNNs are indistinguishable from the real PNN in terms of giant component size, clustering coefficients, and mean number of neighbours, and all are equally robust to vertex removal. These properties are likely due to phonotactics, rather than the definition of the neighbourhood relation or any underlying cognitive constraints.

A promising avenue for further study is the strong assortativity observed on the real PNN relative to the random PNNs, suggesting that there could be principles and mechanisms governing the structure of the lexicons of human languages which enhance the assortativity of the network. Whether these principles operate over milliseconds (i.e. they are caused by patterns of cognitive processing) or generations (i.e. they are caused by patterns of cultural evolution) is a promising question for future research. Replicating these results for languages other than English is also a crucial step in establishing the true nature of PNNs.

Acknowledgements This work has received support under the program “Investissements d’Avenir” launched by the French Government and implemented by ANR with the references ANR-10-LABX-0087 IEC and ANR-10-IDEX-0001-02 PSL.

References

- [1] Arbesman, S., Strogatz, S. H., & Vitevitch, M. S. (2010). The structure of phonological networks across multiple languages. *International Journal of Bifurcation and Chaos*, 20(03), 679–685.
- [2] Blake, I., & Gilchrist, J. (1973). Addresses for graphs. *IEEE Transactions on Information Theory*, 19(5), 683–688.
- [3] Carlson, M. T., Bane, M., & Sonderegger, M. (2011). Global properties of the phonological networks in child and child-directed speech. In *Proceedings of the 35th Boston University Conference on Language Development* (Vol. 1, pp. 97–109). Somerville, MA: Cascadilla Press.
- [4] Chan, K. Y., & Vitevitch, M. S. (2009). The influence of the phonological neighborhood clustering coefficient on spoken word recognition. *Journal of Experimental Psychology: Human Perception and Performance*, 35(6), 1934–1949.

- [5] Chan, K. Y., & Vitevitch, M. S. (2010). Network structure influences speech production. *Cognitive Science*, 34(4), 685–697.
- [6] Deza, M., & Grishukhin, P. (1993). Hypermetric graphs. *The Quarterly Journal of Mathematics Oxford* (2), 44, 399–433.
- [7] Deza, M., & Laurent, M. (1994) ℓ_1 -rigid graphs. *Journal of Algebraic Combinatorics*, 3, 153–175.
- [8] Deza, M., & Shpectorov, S. (1996). Recognition of the ℓ_1 -graphs with complexity $O(nm)$, or football in a hypercube. *European Journal of Combinatorics*, 17(2), 279–289.
- [9] Graham, R. L., & Winkler, P. M. (1985). On isometric embeddings of graphs. *Transactions of the American Mathematical Society*, 288(2), 527–536.
- [10] Gruenenfelder, T. M., & Pisoni, D. B. (2009). The lexical restructuring hypothesis and graph theoretic analyses of networks based on random lexicons. *Journal of Speech, Language, and Hearing Research*, 52(3), 596–609.
- [11] Imrich, W., & Klavžar, S. (1997). Recognizing Hamming graphs in linear time and space. *Information Processing Letters*, 63(2), 91–95.
- [12] Mahowald, K., Dautriche, I., Gibson, E., Christophe, A., Piantadosi, S. T. (In revision). Lexical clustering in efficient language design.
- [13] Newman, M. E. (2003). Mixing patterns in networks. *Physical Review E*, 67(2), 026126.
- [14] Newman, M. E. (2003). The structure and function of complex networks. *SIAM review*, 45(2), 167–256.
- [15] Nusbaum, H. C., Pisoni, D. B., & Davis, C. K. (1984). Sizing up the Hoosier Mental Lexicon: Measuring the familiarity of 20,000 words. *Research on Speech Perception Progress Report, Indiana University*, 10, 357–376.
- [16] Shoemark, P., Goldwater, S., Kirby, J., & Sarkar, R. (2016). Towards robust cross-linguistic comparisons of phonological networks. In *Proceedings of the 14th ACL SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, 110.
- [17] Shpectorov, S. V. (1993). On scale embeddings of graphs into hypercubes. *European Journal of Combinatorics*, 14(2), 117–130.
- [18] Siew, C. S. (2013). Community structure in the phonological network. *Frontiers in Psychology*, 4, 553.
- [19] Stella, M., & Brede, M. (2015). Patterns in the English language: phonological networks, percolation and assembly models. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(5), P05006.
- [20] Vitevitch, M. S. (2008). What can graph theory tell us about word learning and lexical retrieval? *Journal of Speech, Language, and Hearing Research*, 51(2), 408–422.
- [21] Zörnig, P., & Altmann, G. (1983). The repeat rate of phoneme frequencies and the Zipf-Mandelbrot law. *Glottometrika*, 5, 205–211.

Dominance, Deference, and Hierarchy Formation in Wikipedia Edit-Networks

Jürgen Lerner and Alessandro Lomi

Abstract Does co-editing of Wikipedia articles reveal users dominating others? Do these dyadic dominance orderings (if any) lead to a global linear hierarchy among contributing users? In this article we claim that dominance (respectively deference) is revealed by users undoing (respectively redoing) edits of others. We propose methods to turn the history of Wikipedia pages into a dynamic multiplex network resulting from three types of interaction events: dyadic dominance, dyadic deference, and third-party assigned dominance ties. We analyze various local temporal patterns for the different types of ties on a sample of page histories comprising 12,719 revisions by 7,657 unique users. On the dyad level we analyze whether two users tend to agree on a dominance order among them or whether dominated users tend to fight back. On the neighborhood level we analyze various degree effects including whether dominant users tend to dominate in the future and whether subordinate users tend to get dominated. On the triad level we analyze whether users have a preference for transitive closure over cyclic closure of dominance ties. These dynamic patterns shed light on the micro processes that can foster or impede the emergence of a global linear hierarchy.

1 Introduction

The formation of dominance hierarchies is a universal pattern in many human and non-human societies. For instance, experiments with domestic chicken [5, 15] revealed that interaction among two individuals results with overwhelming probability in a clearly dominant and a clearly subordinate one, that dominant (respectively subordinate) individuals tend to dominate (respectively get dominated by) others, and that dominance networks of several individuals tend to be transitive and cycle-

Jürgen Lerner (e-mail: juergen.lerner@gmail.com)✉
University of Konstanz, Germany

Alessandro Lomi (e-mail: alessandro.lomi@usi.ch)
Università della Svizzera italiana, Lugano, Switzerland

free. Experiments on dominance among humans (often denoted by terms like *status*, *reputation*, *prestige*, or *power*) have been performed with small groups (compare [6] and references therein) but empirical studies on hierarchy formation in larger and non-artificial human groups are rare. Collaboration in Wikipedia provides an opportunity to study large-scale, longitudinal, and completely observed data on hierarchy formation in task-oriented human groups. Analyzing hierarchy formation is relevant for understanding Wikipedia since, as any production community, it has to solve the problems of coordination and control. Moreover, acquired high or low status might be a primary source of motivation or frustration of users [18]. However, this paper does not attempt to determine the consequences of successful or failed hierarchy formation but rather analyzes the micro-processes that foster or impede the formation of a global linear hierarchy.

Contributions. In this paper we propose methods to turn the histories of Wikipedia pages into sequences of three types of timestamped and weighted interaction events: dyadic dominance, dyadic deference, and third-party assigned dominance ties. Dyadic dominance ties result from undoing edits and are tentatively interpreted as user *A* claiming: “I (*A*) dominate you (*B*).” Dyadic deference ties result from redoing edits and are tentatively interpreted as *A* claiming: “You (*B*) have high status.” Finally, third-party assigned dominance ties result from user *C* favoring *A*’s edits over *B*’s edits and are tentatively interpreted as *C* claiming: “*A* dominates *B*.” Thus, the difference between dyadic dominance and third-party dominance is whether the dominance from *A* to *B* is claimed by *A* or by a third actor *C*.

We turn these events into a dynamic multiplex network, encoding past interaction among users. Crucially, we aggregate not only the type and weight of events that are actually observed but normalize by the *potential* for such events. We analyze how a tie’s embedding in the network of past events influences the probability of future typed events on it (see Figure 1 for details). This analysis tests the validity of the tentative interpretation of events and reveals which of these types are appropriate or inappropriate for uncovering dominance among users.

2 Background and related work on hierarchy formation and Wikipedia research

Linearity of hierarchies. Dominance hierarchies are universal in groups of many non-human and human species, e. g., [2, 5, 15]. This tendency to form linear hierarchies has often been attributed to advantages in the group’s fitness (cf. [2, 17]); an interesting perspective for our topic: can the success or failure of task-oriented online communities be explained by the (in-)ability to form a hierarchy? Whatever the hypothetical causes or consequences of hierarchy formation, empirical tests of these need ways to assess the degree of linearity in the hierarchical structure of a group. Indices for linearity that have been defined for *tournament graphs* (i. e., graphs in which every undirected dyad $\{A, B\}$ has a dominant and a subordinate node), such as Landau’s *h* or Kendall’s *K*, have been shown to be inappropriate for sparse networks

[16]. Global hierarchy indices for sparse graphs exist (e. g., [14]); alternatively, it has been proposed to measure the linearity of sparse dominance graphs via the relative frequencies of small subgraphs, most notably transitive triads (pointing to linearity) and cyclic triads (pointing to non-linearity) [16, 17].

In this paper we will also consider local configurations but we stress two differences to the two last-mentioned papers. First, we are not analyzing networks of stable dominance ties but dynamic networks of relational events. Thus, instead of counting configurations, we model the probability of current events on a dyad (A, B) as a function of how (A, B) is locally embedded into the network of past events. Second, in networks resulting from the co-editing among Wikipedians there is no reason to assume *a priori* that reciprocated dominance ties are rare. This marks a considerable difference to, say, pecking-networks among chicken where dominance ties are rarely reverted [5]. In Wikipedia, anecdotal evidence, such as the term “edit war” or the “three-revert rule¹”, suggests that at least some users do not accept it when their edits are undone but have a tendency to fight back. Therefore we must start our analysis not with analyzing types of triangles or stars but on the lower dyadic level. Figure 1 illustrates the different network effects considered in this paper.

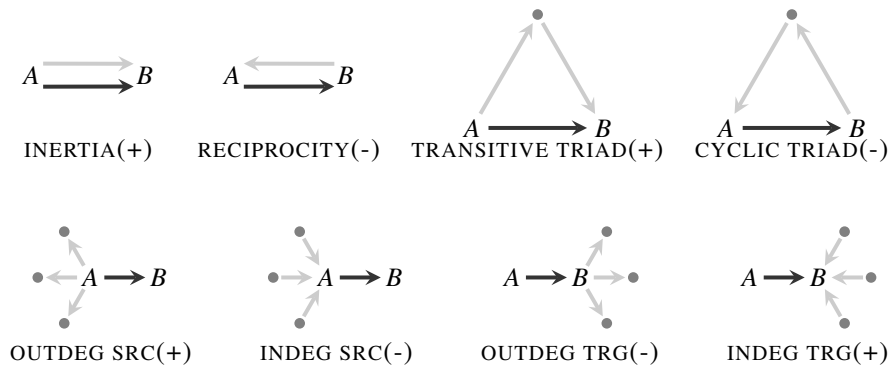


Fig. 1: Local configurations of past dominance events (light gray) explaining future dominance on the tie from A to B (dark gray). A plus sign (+) indicates a hypothetical increase in the probability; a minus sign (-) indicates a hypothetical tendency for decreased dominance probability on (A, B) . All of these hypotheses are derived from the assumption that dominance ties point from higher to lower in the hierarchy. Note that the ties are not binary but have weights between zero and one, as explained in Sect. 3.

Wikipedia research. Wikipedia² is an open, Web-based project to create a user-generated encyclopedia using wiki software [12]. Launched in 2001, Wikipedia is

¹ https://en.wikipedia.org/wiki/Wikipedia:Edit_warring

² www.wikipedia.org

one of the Top-10 most visited websites worldwide³ and is the largest and most popular general reference work on the internet. Its societal relevance, together with the free availability of its complete database, made Wikipedia also a popular case for empirical research and here we can only discuss some of the most closely related previous work. *Reputation* systems for Wikipedians have been proposed, e. g., in [1, 8]. It has been shown, among others, that contributions of users with low reputation are more likely to be undone in the future; this finding corresponds to the hypothesized effect of INDEGREE TARGET in the notation from Figure 1. Other possible patterns in the Wikipedia edit networks are, however, not tested in these two papers, but the largest difference is that we do not seek to define a global reputation index for users but systematically evaluate dynamic local patterns that can foster or hinder the emergence of a linear dominance hierarchy. *Event sequences* (compare [4]) resulting from co-editing Wikipedia articles are analyzed in [7, 9] but none of these papers is specifically about dominance among users (nor about status or reputation of users). *Signed networks* (that is, networks with positive and negative ties) have been defined resulting from co-editing articles ([3]), from votes for or against requests for adminship ([11]), or from both ([13]). Subsequently, these three papers analyze triadic or global patterns confirming or contradicting balance theory and/or status theory in these signed networks. Adding to these previous papers, we evaluate more systematically the consistency of local dynamic patterns with linear hierarchy formation on the dyad level, the neighborhood level (degree effects), and the triadic level. As it has been argued above and will be empirically shown below, the formation of linear hierarchies can be challenged not only with triads but already at a lower level. Last but not least, to the best of our knowledge our paper is the first that also considers third-party assigned dominance ties in which a user C states a dominance order between two different users A and B . The distinction between dyadic dominance and third-party dominance is highly important, since—as we will show in this paper—the latter type of dominance ties is more consistent with linear hierarchy formation.

3 Dominance, deference, and third-party dominance

Edit events. We propose to compute relational events expressing dominance, deference, and third-party dominance by successively comparing the text of subsequent revisions of the same Wikipedia article in a similar way as in previous work, e. g., [1, 3, 8, 13]. As in these papers, we determine for each revision which part of the text is newly added, which is deleted, and which previously deleted text is restored by reverting a deletion. As in previous work, we do not treat it as a text modification if large parts of the text (complete sentences in our case) are just moved or duplicated. As it is usual, we consider a sequence of consecutive revisions by the same user as one revision whose text is that of the last one in the sequence. Authorship of text is maintained at the word level. Note that the same word can appear in different places

³ <http://www.alexa.com/topsites>

in the text and these different instances can have different authors. Augmenting the computation of edit events proposed in [3], we encode the user interaction resulting from it in a more complete way, as explained in the following.

For each word w in the text of each revision we maintain pointers to three potentially different users playing different roles with respect to w :

$$[\text{author}(w), \text{deleter}(w), \text{restorer}(w)] .$$

Here $\text{author}(w)$ is the author who originally added the word w . This pointer is set at the revision when the word is added and is never changed afterward. The pointer $\text{deleter}(w)$ gives the last user who deleted the word. It points to nil when the word is originally added (indicating that no one deleted it so far) and is updated whenever the word is deleted. The pointer $\text{restorer}(w)$ gives the last user who added or restored the word. It is set to the author when the word is originally added but, in contrast to $\text{author}(w)$, the last restorer of a word can change over time when a word is restored after being deleted.

Adding a word, thus, assigns the author of it but creates no interaction events. Interaction events arise when a word is deleted or restored as defined in Figure. 2. Note that we generate a dyadic event only if `active` (i. e., the user who performs the revision) is different from the target of the event and we generate a third-party dominance event only if the active user, the source, and the target are three pairwise different users.

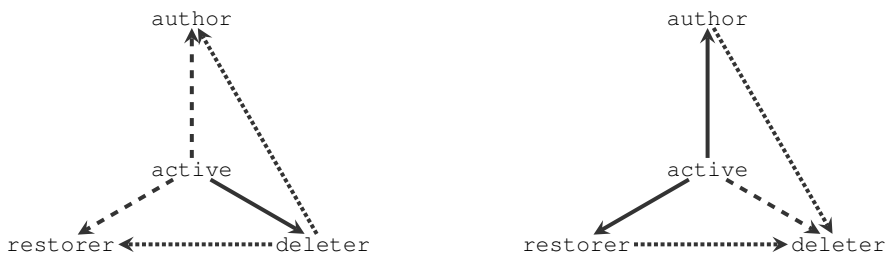


Fig. 2: Edit events resulting from the deletion of a word (*left*) and a word being restored (*right*). Solid lines encode *dyadic deference* events by which the active user re-does the target user’s edit. Dashed lines encode *dyadic dominance* events by which the active user makes the target user’s edit undone. Dotted lines encode *third-party dominance* assignments by which the active user re-does the source user’s edit that has been made undone by the target user. After deleting a word w the user `active` becomes `deleter(w)` and after restoring w user `active` becomes `restorer(w)`. Note that $\text{author}(w)$ is only set when w is originally added and does never change again.

The event potential. While iterating over the revisions of a page we do not only consider events that happen but also the *potential* for such events. More precisely, we keep track for each user B and for each of the dyadic event types x (that is, dyadic dominance and dyadic deference) how many events of type x can have target B .

Likewise, for each ordered pair of different users (A, B) we keep track of the potential for third-party dominance events which a user C (different from A and from B) can assign to the dyad (A, B) .

The network of past events. While iterating over the sequence of revisions of a page, we successively update six functions (called *dyad-level attributes*) defined on ordered pairs (A, B) of different users. Three of these attributes count events of the three types that actually happened on (A, B) and three of them (the *cumulative potentials*) add up the number of events (of the three types) that could have happened on (A, B) at the edit times.

Finally, to describe the past interaction on dyads (A, B) we consider, separately for the three event types, the ratio of actually observed events divided by the cumulative potential for such events.⁴ These ratios are between zero and one (including these borders) and can be interpreted as probabilities: the *past dyadic dominance ratio* on (A, B) is the probability that a randomly chosen word of B that could have been made undone by A during the history of the page is actually undone by A . Similar interpretations apply to *past dyadic deference ratio* and *past third-party dominance ratio*. Henceforward, when we speak of past dominance, deference or third-party dominance, we refer to these ratios.

4 Statistical model

Outcome variables. Whenever a revision r is performed by a user A , then A has a certain potential to initiate events of the two dyadic types to various target users B and A has a certain potential to initiate third-party dominance events on various dyads (B, C) .⁵ The three outcome variables that we consider are the ratios of the number of events actually performed in r divided by the respective potential for such events. Thus, for each event type we use a binomial model where instances are words that can potentially be changed, a “success” instance is such a word that is actually changed in the revision, and a “failure” instance is such a word that is left unchanged. The probability that a potential change occurs is specified in logistic regression models with explanatory variables introduced below.

Explanatory variables. When modeling the probability of change events that could happen in revision r , we use only information about past interaction resulting from revisions that happened strictly before r . These explanatory variables are defined by combinations of three dyadic attributes (past dyadic dominance ratio, past dyadic deference ratio, and past third-party dominance ratio) on the configurations shown in Figure 1. Specifically, for the degree variables we add up the attribute values of all in-coming respectively out-going dyads incident to source respectively target. For the transitive triad variables we sum over all users C (different from A and B) the product

⁴ Here we resolve $0/0$ to be equal to 0, since no event of that type could have happened so far on such a dyad.

⁵ Here we speak of the potential for events in revision r . Note the difference to the cumulative potential used for defining tie-weights in the network of past events.

of the attribute value on (A, C) with the value on (C, B) and take the square-root of this sum. For the cyclic triads we consider the dyads (C, A) and (B, C) accordingly.

To obtain better interpretable explanatory variables we divide them by their standard deviation. With this normalization it is easier to compare the effect sizes of the various variables. Since average probabilities are very close to zero (cf. Table 1), we can interpret the estimated parameters in the following intuitive (not formally correct) way: if we estimated a parameter θ for the variable x when modeling the dyadic dominance probability p , then (hypothetically) increasing x by one standard deviation (that is by 1) multiplies the probability p by $\exp(\theta)$.

Empirical data. We analyzed the histories of a sample of ten articles from the English-language Wikipedia, randomly chosen from the set of articles that have at least 1000 revisions.⁶ In March 2016 there are 56,042 articles (pages in the main namespace that are not redirects) that have at least a thousand revisions. (Altogether there are about 5 million articles; the mean number of revisions per article is just 86.) The ten sampled articles have together 12,719 revisions (disregarding successive revisions by the same user) performed by 7,657 different users. We note that our

Table 1: Number of instances and non-null instances in the analyzed data.

	dyadic dominance	dyadic deference	third-party dominance
no. potential dyads	3,126,047	1,753,160	4,852,052
no. non-null dyads	37,823	21,411	21,335
dyad-density	1.21%	1.22%	0.44%
no. potential words	361,673,769	359,365,077	348,420,292
no. changed words	1,738,728	785,233	783,190
word-change density	0.48%	0.22%	0.23%

number of observations is not just ten since the unit of analysis is not the page but the dyadic event. Table 1 gives the number of dyad-timepoints on which there could have happened an event of the various types, the number of actual dyadic events, the number of words that could have been modified, and the number of actual word modifications. The approach to analyze 10 random pages (rather than just one) has been chosen since it reduces the likelihood of accidentally analyzing a page with an exceptional structure. The restriction to pages with at least a thousand revisions is motivated by the consideration that hierarchy formation takes some time and also a number of users that is not too small. What blows up the runtime of our analysis is that we consider not only the actually occurring events but also those that could have happened. However, we strongly believe that this is necessary since an observation such as “user A deleted 10 of user B ’s words” is meaningless if we disregard how

⁶ These turned out to be the pages: Balika Vadhu; Ganymede (moon); Greed; Jay Park; List of Hollyoaks locations; Mothra; Pea; Shiv Sena; Swimsuit; and The Third Man.

many of B 's words user A did not touch and/or if we disregard all the other users with which A potentially could have interacted but did not. The results reported in the next section have been estimated to maximize the joint likelihood of all events from all sampled pages.

5 Results and discussion

Dyad-level effects. Table 2 reports logistic regression parameters explaining the probability of dyadic dominance by past interaction on the same and the reverse dyad. In the first model, we observe that past dyadic dominance on (A, B) increases the probability of future dyadic dominance on (A, B) ; thus, actors continue to dominate their subordinates. However, we see that past dyadic dominance on the reverse dyad (B, A) also increases the probability of dyadic dominance on (A, B) ; thus, subordinate actors have a tendency to fight back which is a hindrance to hierarchy formation. Likewise, we see that past deference on (A, B) reduces the probability of dyadic dominance on (A, B) (as expected). However, past deference on (B, A) also reduces the probability of dyadic dominance on (A, B) ; this makes the interpretation that deference goes from lower to higher in the hierarchy questionable.

Table 2: Explaining dyadic dominance by past dyadic dominance and dyadic deference on the same dyad.

	dyad model	dyadic inertia	dyadic reciprocity
(Intercept)	-5.427 (0.001)***	-5.427 (0.001)***	-5.427 (0.001)***
dyadic dominance inertia	0.222 (0.000)***	0.115 (0.000)***	
dyadic deference inertia	-0.288 (0.002)***	-0.071 (0.003)***	
dyadic dominance reciprocity	0.060 (0.000)***		-0.064 (0.000)***
dyadic deference reciprocity	-0.093 (0.000)***		0.030 (0.001)***
undirected dyadic dominance		0.127 (0.000)***	0.262 (0.000)***
undirected dyadic deference		-0.243 (0.001)***	-0.321 (0.003)***
AIC	17,531,308.231	17,531,308.231	17,531,308.231
Num. obs.	3,126,047	3,126,047	3,126,047

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

Looking more closely at the parameter sizes, we see that a dyadic dominance event on (A, B) has two effects. First it increases the future hostility (likelihood of dominance events) on (A, B) and on (B, A) . This is consistent with a structural balance interpretation of negative ties (and inconsistent with a status interpretation) and has also been found by Leskovec et al. [11] who analyzed voting behavior of Wikipedians. A second effect, however, is that a dyadic dominance event on (A, B) increases the future dominance on (A, B) more than on (B, A) , thereby increasing

the relative dominance of (A, B) over (B, A) . This second effect becomes more transparent if we control for the increase in dominance activity on both dyads (A, B) and (B, A) by defining a variable *undirected dyadic dominance* which is the sum of dyadic dominance inertia and dyadic dominance reciprocity (normalized to standard deviation one). In the second and third model in Table 2 we see that, controlling for the undirected increase in dominance activity, a dominance event on (A, B) increases the future dominance probability on (A, B) more than expected and that it increases the future dominance probability on (B, A) less than expected. A similar result is obtained for dyadic deference, where a deference event on (A, B) *decreases* the future dominance probability on (A, B) more than expected and that on (B, A) less than expected. We note that the three models in Table 2 are equivalent since their variables are linear transformations of each other.

Summarizing this, a dyadic dominance event on (A, B) has two effects: a structural balance effect increasing the hostility level on the undirected dyad $\{A, B\}$ and a hierarchical effect that shifts the relative dominance towards the direction (A, B) . It is likely that the experimentally found anti-reciprocity of dominance events among chicken (e. g., [5, 15]) is due to the small network size. In larger and therefore sparser networks it is likely that reciprocation of acts of dominance, albeit rare, would occur with a higher probability than the low baseline probability of interacting at all.

We make similar findings when estimating the probability of dyadic deference by dyadic effects (with the understanding that deference hypothetically points from lower to higher). These results are not reported in this paper.

Table 3: Explaining third-party dominance by past third-party dominance on the same dyad.

	dyad model	dyadic inertia	dyadic reciprocity
(Intercept)	-6.196 (0.001)***	-6.196 (0.001)***	-6.196 (0.001)***
tp dominance inertia	0.379 (0.000)***	0.391 (0.001)***	
tp dominance reciprocity	-0.022 (0.001)***		-0.740 (0.001)***
undirected tp dominance		-0.025 (0.001)***	0.814 (0.001)***
AIC	9,721,545.325	9,721,545.325	9,721,545.325
Num. obs.	4,852,052	4,852,052	4,852,052

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

Table 3 reports logistic regression parameters explaining the probability of third-party dominance by past interaction on the same and the reverse dyad. In contrast to dyadic dominance, we see that third-party dominance is clearly anti-reciprocal: controlling for the undirected increase in the event probability is here not necessary although it strengthens the anti-reciprocity. This means that if a different user C states that A dominates B , then the probability that C (or any other user different from A and

B) later reverses this order decreases. Thus, third-party assigned dominance is more consistent with the hierarchical interpretation than dyadic dominance. Apparently bystanders can judge the dominance order among *A* and *B* more reliably than *A* or *B* themselves.

Neighborhood-level effects (degree effects). We estimated models explaining the probability of dyadic dominance on (*A*, *B*) by past interaction on edges incident to *A* (source) and *B* (target). For space limitations, the estimated parameters are not reported in this paper and we will only summarize the main findings. We find some effects consistent with the hierarchical interpretation, such as a positive effect of *dominance outdegree source* and *dominance indegree target*. However, we can also find effects inconsistent with this interpretation, such as a positive effect of *dominance outdegree target* (which implies that dominant users tend to get dominated). As in the case of dyad effects, the effects of the degree variables (for dominance and for deference) become consistent with the hierarchy-interpretation once we control for the *undirected* degrees. We also controlled for the dyadic effects from Table 2 in the degree model which did not change the findings qualitatively. We further estimated the probability of dyadic deference events by degree effects. These findings differ qualitatively from those obtained for the dominance probability (whether or not we control for the undirected degrees) and make the interpretation of dyadic deference pointing from subordinate to dominant more questionable.

We also estimated degree-models for third-party dominance (not reported in this paper). Most effects in this model are consistent with the hierarchical interpretation of third-party dominance ties. The exception is a positive effect of *indegree source* which suggests that subordinates are more likely to dominate in the future. As for dyadic dominance, controlling for the undirected degrees brings all effects in accordance with the hierarchical interpretation. Controlling for the dyadic effects from Table 3 in the degree model yields qualitatively the same findings.

Triad-level effects. Table 4 reports logistic regression parameters explaining the probability of dyadic dominance on (*A*, *B*) by past interaction on two-paths of the form (*A*, *C*), (*C*, *B*), forming a transitive triad, and on two-paths of the form (*B*, *C*), (*C*, *A*), forming a cyclic triad. The first model reveals that the embedding of (*A*, *B*) in a dominance two-path increases the probability of a dominance event on (*A*, *B*), irrespective of whether the resulting triad is transitive or cyclic. Controlling for the increase in dominance activity caused by a dominance two-path in any direction (*dominance triplet*) reveals a preference for transitive over cyclic closure of dominance ties—consistent with the formation of a linear hierarchy. Similar effects result from two-paths of deference ties. Controlling for dyad effects and degree effects (not reported in this paper), however, does *not* keep these effects stable.

Table 5 reports logistic regression parameters explaining the probability of third-party dominance on (*A*, *B*) by past interaction on two-paths of the form (*A*, *C*), (*C*, *B*), forming a transitive triad, and on two-paths of the form (*B*, *C*), (*C*, *A*), forming a cyclic triad. The first model reveals that indirect third-party dominance ties decrease the probability of third-party dominance on the dyad (*A*, *B*) irrespective of the direction of these two-paths. For transitive triplets, this contradicts the hierarchical interpretation

Table 4: Explaining dyadic dominance by past dyadic dominance and dyadic deference on transitive and cyclic two-paths.

	triad model	transitive triad	cyclic triad
(Intercept)	-5.264 (0.001)***	-5.264 (0.001)***	-5.264 (0.001)***
transitive dominance triplet	0.149 (0.001)***	0.049 (0.001)***	
transitive deference triplet	-0.362 (0.002)***	-0.045 (0.003)***	
cyclic dominance triplet	0.027 (0.000)***		-0.013 (0.000)***
cyclic deference triplet	-0.135 (0.001)***		0.019 (0.001)***
dominance triplet		0.105 (0.001)***	0.156 (0.001)***
deference triplet		-0.355 (0.002)***	-0.405 (0.003)***
AIC	18,958,234.850	18,958,234.850	18,958,234.850
Num. obs.	3,126,047	3,126,047	3,126,047

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

Table 5: Explaining third-party dominance by past third-party dominance on transitive and cyclic two-paths.

	triad model	transitive triad	cyclic triad
(Intercept)	-5.403 (0.001)***	-5.403 (0.001)***	-5.403 (0.001)***
transitive tp dominance triplet	-0.024 (0.001)***	1.330 (0.002)***	
cyclic tp dominance triplet	-1.792 (0.003)***		-1.760 (0.003)***
tp dominance triplet		-2.259 (0.004)***	-0.040 (0.001)***
AIC	10075300.988	10,075,300.988	10,075,300.988
Num. obs.	4,852,052	4,852,052	4,852,052

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

but is consistent with a structural balance interpretation of dominance ties (an enemy of an enemy is not an enemy). When we control for the dominance-reducing effect of undirected two-paths, we find a preference for transitive over cyclic closure (consistent with the hierarchical interpretation). As for dyadic dominance, controlling for dyad and degree effects (not reported in this paper) does *not* keep these triadic effects stable.

6 Conclusion

In this paper, we proposed methods to derive three types of interaction events from co-editing Wikipedia articles. We analyzed whether local dynamic patterns for

these events are consistent with a linear dominance hierarchy among the users. The analysis in this paper revealed that past events can have two distinct effects on future interaction: on one hand on the frequency of events on the undirected dyad $\{A, B\}$ and on the other hand on the relative dominance of (A, B) over (B, A) . The effects on the undirected dyads are often more consistent with a structural balance interpretation of dominance events as revealing negative ties. The effects on the directed dyads are often more consistent with a hierarchical interpretation of dominance events. This finding is similar to one made in [11] where voting behavior among Wikipedians was analyzed. We also showed that the effect on the event frequency can obfuscate effects on the hierarchical ordering. This finding is similar to one made in [10] where effects on the interaction frequency were separated from effects influencing the sign of ties. The analysis in our paper also revealed that the three different types of events show different levels of consistency with linear dominance hierarchies. Most notably, third-party assigned dominance was the only event type that is anti-reciprocal, irrespective of whether we control for a change in the interaction frequency or not. On the other hand, dyadic deference was the most inconsistent with the hierarchical interpretation. A promising approach for future research is to link patterns of (failed or successful) hierarchy formation with properties of the page, such as article quality. This would need a larger sample of separately analyzed pages that show variation in their hierarchical structure and in quality.

Acknowledgements This work has been supported by Swiss National Science Foundation (FNS Project Nr. 100018_150126) and Deutsche Forschungsgemeinschaft (DFG Grant Nr. LE 2237/2-1).

References

- [1] Adler, B.T., de Alfaro, L.: A content-driven reputation system for the Wikipedia. In: Proc. 16th Intl. Conf. WWW, pp. 261–270 (2007)
- [2] Boyce, W.T., Obradović, J., Bush, N.R., Stamperdahl, J., Kim, Y.S., Adler, N.: Social stratification, classroom climate, and the behavioral adaptation of kindergarten children. *Proceedings of the National Academy of Sciences* **109**(Supplement 2), 17168–17173 (2012)
- [3] Brandes, U., Kenis, P., Lerner, J., van Raaij, D.: Network analysis of collaboration structure in Wikipedia. In: Proc. 18th Intl. Conf. WWW (2009)
- [4] Butts, C.T.: A relational event framework for social action. *Sociological Methodology* **38**(1), 155–200 (2008)
- [5] Chase, I.D.: Dynamics of hierarchy formation: the sequential development of dominance relationships. *Behaviour* **80**(3), 218–239 (1982)
- [6] Fişek, M.H., Berger, J., Norman, R.Z.: Participation in heterogeneous and homogeneous groups: A theoretical integration. *American Journal of Sociology* pp. 114–142 (1991)
- [7] Iba, T., Nemoto, K., Peters, B., Gloor, P.A.: Analyzing the creative editing behavior of Wikipedia editors: Through dynamic social network analysis. *Procedia-Social and Behavioral Sciences* **2**(4), 6441–6456 (2010)
- [8] Javanmardi, S., Lopes, C., Baldi, P.: Modeling user reputation in wikis. *Statistical Analysis and Data Mining* **3**(2), 126–139 (2010)
- [9] Keegan, B.C., Lev, S., Arazy, O.: Analyzing organizational routines in online knowledge collaborations: A case for sequence analysis in CSCW. In: Proc. 19th ACM Conf. Computer-Supported Cooperative Work & Social Computing, pp. 1065–1079. ACM (2016)

- [10] Lerner, J.: Structural balance in signed networks: Separating the probability to interact from the tendency to fight. *Social Networks* **45**, 66–77 (2016)
- [11] Leskovec, J., Huttenlocher, D., Kleinberg, J.: Signed networks in social media. In: Proc. SIGCHI Conf. Human Factors in Computing Systems, pp. 1361–1370. ACM (2010)
- [12] Leuf, B., Cunningham, W.: *The Wiki Way*. Addison-Wesley (2001)
- [13] Maniu, S., Cautis, B., Abdessalem, T.: Building a signed network from interactions in Wikipedia. In: Proc. Databases and Social Networks, pp. 19–24. ACM (2011)
- [14] Mones, E., Vicsek, L., Vicsek, T.: Hierarchy measure for complex networks. *PloS one* **7**(3), e33799 (2012)
- [15] Schjelderup-Ebbe, T.: Beiträge zur Sozialpsychologie des Haushuhns. *Zeitschrift für Psychologie* **88**, 225–252 (1922)
- [16] Shizuka, D., McDonald, D.B.: A social network perspective on measurements of dominance hierarchies. *Animal Behaviour* **83**(4), 925–934 (2012)
- [17] Shizuka, D., McDonald, D.B.: The network motif architecture of dominance hierarchies. *Journal of The Royal Society Interface* **12**, 20150080 (2015)
- [18] Willer, R.: Groups reward individual sacrifice: The status solution to the collective action problem. *American Sociological Review* **74**(1), 23–43 (2009)

Part II

Network Measures

Identifying Influential Spreaders by Graph Sampling

Nikos Salamanos, Elli Voudigari and Emmanuel J. Yannakoudakis

Abstract The complex nature of real world networks is a central subject in several disciplines, from Physics to computer science. The complex network dynamics of peers communication and information exchange are specified to a large degree by the most efficient spreaders - the entities that play a central role in various ways such as the viruses propagation, the diffusion of information, the viral marketing and network vulnerability to external attacks. In this paper, we deal with the problem of identifying the influential spreaders of a complex network when either the network is very large or else we have limited computational capabilities to compute global centrality measures. Our approach is based on graph sampling and specifically on *Rank Degree*, a newly published graph exploration sampling method. We conduct extensive experiments in five real world networks using four centrality metrics for the nodes spreading efficiency. We present strong evidence that our method is highly effective. By sampling 30% of the network and using at least two out of four centrality measures, we can identify more than 80% of the influential spreaders, while at the same time, preserving the original ranking to a large extent.

The original version of this chapter was revised. An erratum to this chapter can be found at [10.1007/978-3-319-50901-3_66](https://doi.org/10.1007/978-3-319-50901-3_66)

Nikos Salamanos

Athens University of Economics and Business, 76 Patission Str. GR10434 Athens Greece, e-mail: salaman@aueb.gr

Elli Voudigari

Athens University of Economics and Business, 76 Patission Str. GR10434 Athens Greece, e-mail: elliv@aueb.gr

Emmanuel J. Yannakoudakis

Athens University of Economics and Business, 76 Patission Str. GR10434 Athens Greece, e-mail: eyan@aueb.gr

© Springer International Publishing AG 2017

H. Cherifi et al. (eds.), *Complex Networks & Their Applications V*,

Studies in Computational Intelligence 693,

DOI 10.1007/978-3-319-50901-3_9

1 Introduction

Understanding spreading process in real world complex networks is of high importance due to the variety of applications that they occur, such as the acceleration of information diffusion, the control of the spread of a disease and the improvement of the resilience of networks to external attacks.

Key role to spreading dynamics plays the heterogeneity of nodes in terms of *spreading efficiency*. High spreading efficient nodes are called *influential spreaders*, representing the nodes that are more likely to spread information or a virus in a large part of the network. Therefore, thorough research has been realized in order to connect the topological properties of network nodes with their spreading efficiency.

In this paper, we deal with the problem of identifying the influential spreaders of a complex network when we are not able to analyze directly the whole network, either because of its large size or of our limited computational resources which are necessary for estimating global centrality measures or other advanced nodes properties.

Our approach is based on graph sampling, the problem of selecting a small sub-graph which will preserve the topological properties of the original graph. In our case, the central question is whether the top-k spreaders in the samples correspond to the top-k spreaders in the original graph. Thus, a sampling method could be served effectively as an *influential spreaders identifier* if and only if: (a) the fraction of top-k common nodes in the samples and in the graph is on average sufficiently large and (b) the rankings of these nodes in the samples are close to the original ranking in the graph.

We address this question using *Rank Degree* [18], a graph exploration sampling method which as proven outperforms other well known methods such as *Forest Fire* and *Frontier sampling* [11, 10, 14].

We conduct extensive experiments in five real world networks using four centrality metrics in order to rank the nodes, with respect to spreading efficiency. In order to emphasize the efficiency of Rank Degree, we compare our method with that of Forest Fire. The results show that Forest Fire is inadequate in identifying the best spreaders, while our method is highly effective. Studying the samples of Rank Degree, we are able to identify in every network, at least 80% of the influential spreaders by sampling 30% of the network, using at least two out of four centrality measures.

Finally, and more importantly, in four out of five networks, the rank correlation between the top-k nodes in the samples and the top-k nodes in the original graph is very large.

The rest of the paper is organized as follows. Sect. 2 describes the related work. Sect. 3 presents our method. Sect. 4 describes the experimental analysis and provides information on the methods and datasets used and Sect. 5 concludes the paper.

2 Related Work

The problem of identifying the influential spreaders in a network is a central subject in complex networks analysis and therefore, several approaches have been proposed in the literature.

Kitsak *et al.* [9] proposed the k -shell decomposition method [15, 16] as an *influential spreaders identifier*, showing that the k -core values constitute a more reliable measure than *degree centrality* and *betweenness centrality*. One of the core results is that the placement of a node (node global property) is more important than its degree (node local property). Two nodes with the same degree but different placement, where the one is connected with the periphery of the network and the other with the innermost core will not have equal spreading efficiency. Thus, highly connected nodes are not always the best spreaders, while less connected nodes but well connected with the core of the network may strongly affect the spreading process. In addition, Zeng *et al.* [19] investigated the limitations of the k -shell method and they proposed a mixed degree decomposition procedure which performs more accurately than the k -shell approach.

Chen *et al.* [2] proposed the *local centrality*, a semi-local centrality measure as a tradeoff between the degree centrality (local measure) and the computationally complex betweenness and closeness (the global measures). They showed that local centrality is more effective to identify influential nodes than the degree centrality.

LeaderRank [13] is a ranking algorithm for identifying influential nodes in directed social networks. LeaderRank is a parameter-free random walk algorithm analogous to PageRank [1]. Moreover, Li *et al.* [12] proposed a weighted variation of Leader Rank which outperforms LeaderRank. Furthermore, in [3] the authors introduced *ClusterRank* a local ranking algorithm for directed graphs that takes into account the nodes *clustering coefficient* and they proved that ClusterRank outperforms other approaches such as LeaderRank.

3 The Rank Degree Method

Algorithm 1 presents briefly the *Rank Degree (RD)* sampling method. *RD* is a graph exploration sampling algorithm which outperforms several other well known approaches. A detailed analysis of the algorithm is out of the scope of this paper and we refer to [18] where the authors studied thoroughly the properties and the efficiency of the algorithm.

The main characteristic of the method is that the graph traverse is based on a deterministic selection rule, the ranking of nodes according to their degree values (see Steps 9-10). The algorithm is specified by two parameters: (a) the number s of the initial starting nodes (seeds) and (b) the parameter ρ which defines the top- k , that is, the selected fraction of nodes from each ranking list. Hence, we use the notation $RD(\rho)$. The extreme case is for top- k with $k=1$, in other words when we

Algorithm 1 Rank Degree Algorithm

```

1: Set parameters: (i)  $s$ : number of initial seeds, (ii)  $\rho$  (see Step-10), (iii) target sample size  $x$ 
2: Input: undirected graph  $G(V, E)$ 
3: Output: sample of size  $x$ 
4: Initialization:  $\{Seeds\} \leftarrow s$  nodes selected uniformly at random
5:  $Sample \leftarrow \emptyset$ 
6: while sample size < target size  $x$  do
7:    $\{New\ Seeds\} \leftarrow \emptyset$ 
8:   for  $\forall w \in \{Seeds\}$  do
9:     Rank  $w$ 's friends based on their degree values
10:    Selection rule:
      (i)  $RD(max)$ : select the max degree (top-1) friend of  $w$ 
      (ii)  $RD(\rho)$ : select the top- $k$  friends of  $w$ , where  $k = \rho \cdot (\#friends(w))$ ,  $0 < \rho \leq 1$ 
11:    Update the current sample with the selected edges ( $w$ ,  $friend(w)$  on the top- $k$ ) along
      with the symmetric ones
12:    Add to  $\{New\ Seeds\}$  the top- $k$  friends of  $w$ 
13:   end for
14:   Update graph  $G$ : delete from the graph all the currently selected edges
15:    $\{Seeds\} \leftarrow \{New\ Seeds\}$ 
      If  $\{New\ Seeds\} = \emptyset$  then repeat Step-4 (random jump)
16: end while

```

select only one node from each ranking list - that node having the maximum degree. For simplicity, we refer to this case as $RD(max)$.

The algorithm, starting from s initial nodes, performs s parallel graph traverses. At each time step, the number of visited nodes (current seeds) varies and depends on the set of selected nodes at the previous time step.

As referred to, in [18], the algorithm generates the most representative samples for $RD(max)$ and $RD(0.1)$, i.e. when we select either the top-1 or the top-10% from the ranking lists. In this paper, we concentrate our analysis to $RD(max)$ studying its performance with respect to influential spreaders.

4 Experimental Analysis

4.1 Methods

Sampling: Apart from our method, RD , we study the *Forest Fire (FF)*, a well known sampling method introduced by Leskovec *et al.* [11]. FF starts from a randomly selected node (seed) w and at each step, the algorithm moves from the current set of seeds to the next one as follows: from each node w in the set of current nodes, a random number x is generated which is geometrically distributed with mean $p_f(1 - p_f)$. The parameter p_f is called *forward burning probability* which is set to 0.7. Then, x outgoing edges are selected from the set of w 's outgoing edges. The end nodes of the selected edges constitute the next set of current nodes. At each step,

the visited nodes are considered as burned and are removed from the graph. Hence, they cannot be traversed for a second time. Finally, the process is repeated until a sample of the requested size is reached.

Spreading efficiency: In the absence of ground truth information with regard to nodes spreading efficiency, several approaches have been proposed in the literature such as the *Linear Threshold* and *Independent Cascade* models [7], as well as the basic epidemic models *Susceptible Infected Recovered (SIR)* and *Susceptible Infectious Susceptible (SIS)* [9, 2] which tend to simulate the spreading process in a graph.

In this paper, we use local and global topological properties, centrality measures, in order to estimate the nodes spreading efficiency in the original graph and in the samples: (a) *k-core decomposition*, a subgraph with nodes of degree at least k (on the subgraph). *k-shell*: the set of nodes that belong to the k -core but not to the $k+1$ -core. For the rest of the paper, when we refer to nodes k -core values we imply the max k -shell that these nodes belong to, (b) *degree centrality*, (c) *betweenness centrality* and (d) *closeness centrality* [5].

It has been proved that most of the centrality measures are positive correlated [17] and also that some measures are less effected by sampling [4].

Sampling evaluation: We study the efficiency of the sampling methods with regard to node influences using two measures:

(a) *OSim* [6], an object similarity measure (in our case the objects are the nodes), the overlap between the elements of two ranking lists A and B (each of size k), without taking into account their ordering. It is defined as $OSim(A, B) = \frac{|A \cap B|}{k}$. In our case, the lists A and B correspond to the ranking lists $r_G(top-k)$ and $r_S(top-k)$ which are computed as follows: for a given centrality measure we calculate the nodes centrality values for both the original graph G as well as each of the collected samples S and we rank the nodes accordingly (in descending order) creating the ranking lists r_G and r_S . Then, for a given k , we create the $r_G(top-k)$ and $r_S(top-k)$ collecting the top- k nodes of the ranking lists r_G and r_S .

(b) *Kendall tau* [8], the well known rank correlation coefficient measure, with which we measure the relative ordering between all pair of nodes in the two ranking lists $r_G(top-k)$ and $r_S(top-k)$.

4.2 Data and Sampling Setup

We evaluate the efficiency of $RD(max)$ as influential spreaders identifier in five real world datasets, two of small and three of medium graph size (Table 1). We restrict our analysis to undirected graphs, therefore we transform the directed graphs (*wiki-Vote* and *p2p-Gnutella30*) to undirected, by applying to each edge the symmetric one. In addition, we study the efficiency of FF - a well known sampling algorithm which, contrary to RD , inadequately identifies the most influential nodes, even if it

Table 1 Datasets

Graph	egoFacebook	wiki-Vote	CA-CondMat	p2p-Gnutella30	Email-Enron
Description	Ego-net	Wiki-net	Collaboration Net.	P2P Net.	Comm. Net.
Type	Undirected	Directed	Undirected	Directed	Undirected
# Nodes	4039	7115	23133	36682	36692
# Edges	88234	103689	93497	88328	183831

is producing representative samples with regard to some topological properties of the graph.

For each dataset and each method separately, we collect 40 samples, per sample size, where the sample sizes are 10%, ..., 50%. In all experiments, the number of initial seeds is defined by the 1% of the target sample size. For instance, for a given graph G with 2000 nodes and target sample size 10%, the number of initial seeds is 2. Moreover, we compute the OSim and Kendall tau for each top-k interval separately. Therefore, we define two top-k intervals, the small top-k, where $k \in [0.001, 0.01]$ (i.e. one per mill to one percent) as well as the medium top-k, where $k \in [0.01, 0.1]$ (i.e. 1% to 10%)

4.3 Results

4.3.1 Effectiveness of Rank Degree

Top-k similarity (OSim): For a given graph G , top-k and centrality measure, we calculate the OSim between the top-k nodes in G and the top-k nodes in each of the 40 samples separately.

Fig. 1 and Fig. 2 present the average OSim for $RD(max)$ samples, of the small and medium size graphs. Specifically, for each graph, for each top-k interval, and for each sample size, we plot the average OSim values of the 40 samples, for each centrality measure separately. The results for small and medium top-k (i.e. $k \in [0.001, 0.01]$ and $k \in [0.01, 0.1]$) are given in separate plots. For the sake of clarity, only the sample sizes 10% and 30% are shown.

We observe that, in *egoFacebook* the samples size 30% maintain at least the 80% of influential spreaders in terms of k -core and degree centrality for *small* top-k (Fig. 1(a)), while for *medium* top-k, the corresponding OSim values are larger than 90% (Fig. 1(b)). Moreover, from Fig. 1(c) (*wiki-Vote*), it is clear that all centrality OSim values are higher than 70% for all sample sizes. In medium top-k (Fig. 1(d)) and for samples size 30%, the degree centrality and k -core have the largest OSim values where in some cases are close to 100%.

In Fig. 2(a) (*CA-CondMat*), we can see that for small top-k, degree centrality and closeness centrality are close to 80% with betweenness and k -core following. The results are similar for medium top-k (Fig. 2(b)).

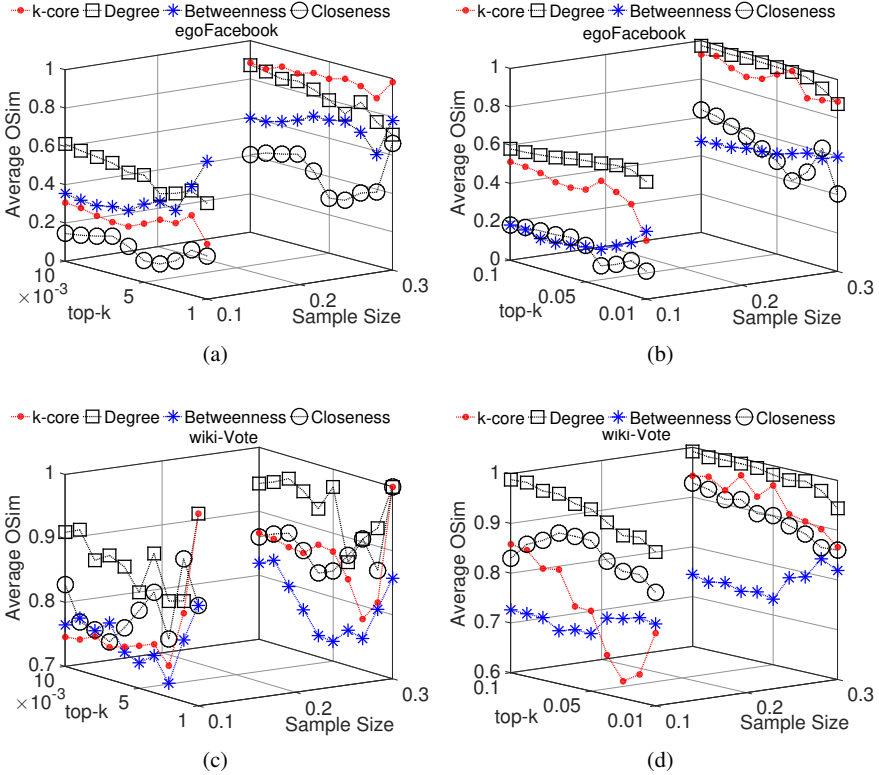


Fig. 1 Average OSim per top-k. Small size graphs

In the case of *p2p-Gnutella30* (Fig. 2(c)), *k*-core comes first for sample sizes 10% and 30% with closeness, degree centrality and betweenness following. For medium top-*k*, three out of four centrality measures have OSim values larger than 80% (Fig. 2(d)).

In *Email-Enron* and small top-*k*, three out of four centrality measures have OSim values larger than 80%. In almost all sample sizes and top-*k* intervals, the OSim for *k*-core is close to 100% (Fig. 2(e)). Finally, the results for medium top-*k* and samples size 30%, three out of four centrality measures have OSim values larger than 90% (Fig. 2(f)).

Ranking similarity (Kendall tau): For a given graph *G*, top-*k* and centrality measure, we apply the Kendall tau on the ranking values of the common nodes between the top-*k* nodes in the graph *G* and in a given sample *S*. Specifically, consider two ranking lists $r_G(top - k)$ and $r_S(top - k)$. First, we compute the intersection $R = r_G(top - k) \cap r_S(top - k)$. Then, we define the $R_G(top - k)$ and $R_S(top - k)$ which contain only the ranking values from $r_G(top - k)$ and $r_S(top - k)$ that corre-

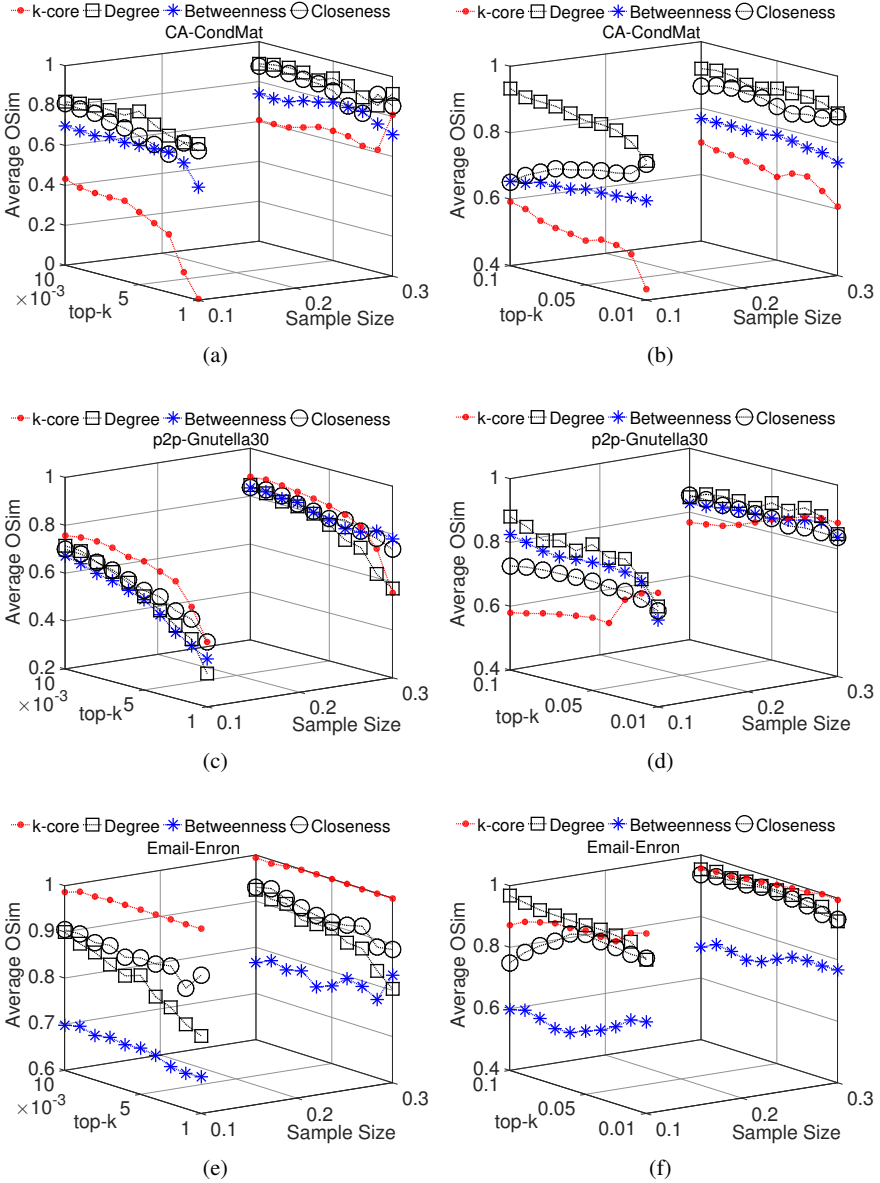


Fig. 2 Average OSim per top-k. Medium size graphs

spond to the nodes in R . Finally, we compute the Kendall tau of $R_G(top - k)$ and $R_S(top - k)$.

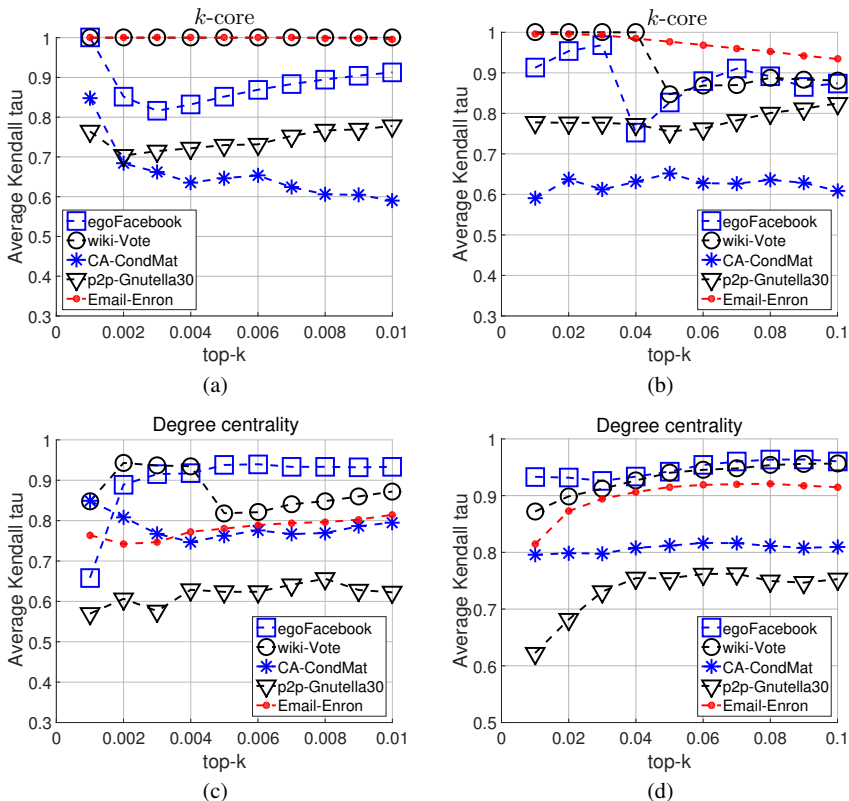


Fig. 3 Ranking similarity: Average Kendall tau per top-k. Samples size 30%

Fig. 3 presents the average Kendall tau values for k -core and degree centrality for small and medium top-k and samples size 30%.

We observe that in four out of five datasets the average Kendall tau values are large, at least 0.7. Thus, there is a large positive correlation between the ordering of the top-k nodes in the samples and the top-k nodes in the original graph.

For instance, in *wiki-Vote* and *Email-Enron*, for small top-k and top-k in $[0.01, 0.4]$, the Kendall’s tau values are almost equal to one (Fig. 3(a) and Fig. 3(b)). Moreover, in every top-k, the samples from all datasets except *CA-CondMat* preserve strongly the relative ordering of the top-k nodes.

In the case of degree centrality, the results are similar. For instance, in four out of five datasets and for any interval of medium top-k, the average Kendall values are at least 0.8 (Fig. 3(d)).

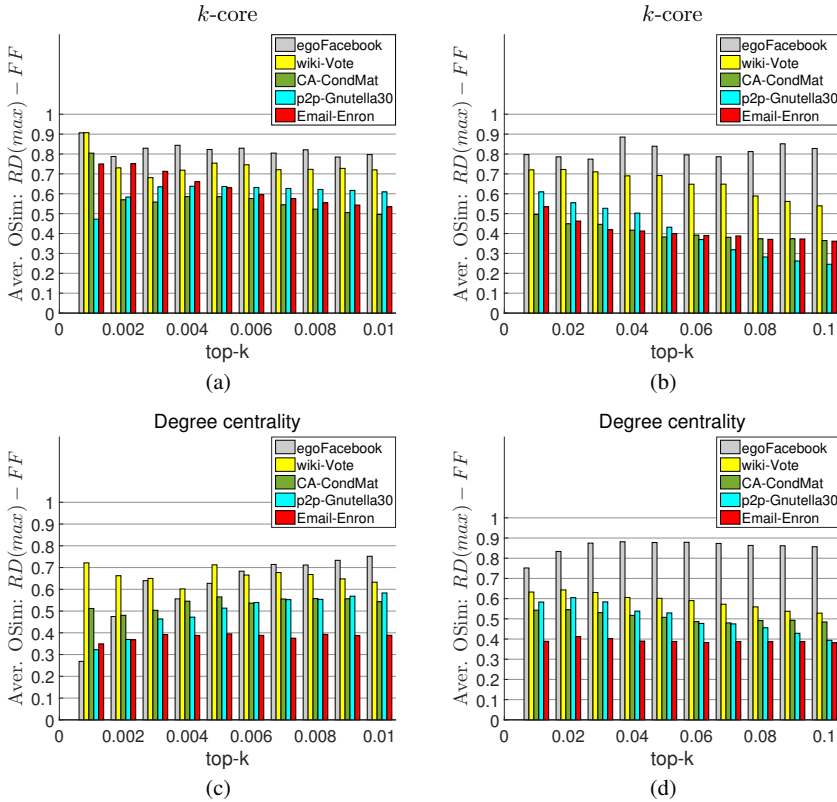


Fig. 4 Comparison of $RD(max)$ and FF : average OSim $RD(max)$ minus average OSim FF per top-k. Samples size 30%

4.3.2 Rank Degree vs Forest Fire

We conclude the analysis comparing our method with the Forest Fire (FF). For each top-k and for each sample size, we compute the difference between the average OSim of $RD(max)$ and the average OSim of FF. We present the results only for k -core and degree centrality, as well as for samples size 30%. The results for the other sample sizes and centrality measures are similar, hence we omit the plots.

Observing the Fig. 4 and taking into account Fig. 1 and Fig. 2, where we present the average OSim between the original graph and all 40 samples, we conclude the following.

In both small and medium datasets and for every top-k, the difference of OSim values in terms of k -core and degree centrality is always positive. The range of difference is roughly between 0.3 to 0.9 which shows that RD is more efficient than FF as an influential nodes identifier.

5 Conclusion

In this paper, we proposed a graph sampling approach to the problem of identifying the influential spreaders in a complex network. Our approach is based on graph sampling and specifically on Rank Degree, an efficient graph exploration sampling algorithm. We experimentally analyzed the proposed method using several centrality measures and studying five real world networks. The analytical experiments demonstrate that our method can identify, with high accuracy, a large fraction of the most influential nodes along with their original ranking in the whole graph. In future, we intend to extend our analysis applying the *SIR* and *SIS* epidemic models that will serve as ground truth information on the spreading efficiency of nodes. More specifically, we will investigate the correlation between the centrality measures and the spreading efficiency of nodes, as defined by the epidemic models in the original graph and in Rank Degree samples.

Acknowledgements We thank Kyriaki Chryssaki for her helpful comments on the final manuscript.

References

1. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems* **30**(1), 107 – 117 (1998)
2. Chen, D., Lu, L., Shang, M.S., Zhang, Y.C., Zhou, T.: Identifying influential nodes in complex networks. *Physica A: Statistical Mechanics and its Applications* **391**(4), 1777 – 1787 (2012)
3. Chen, D.B., Gao, H., Lu, L., Zhou, T.: Identifying influential nodes in large-scale directed networks: The role of clustering. *PLoS ONE* **8**(10), 1–10 (2013)
4. Costenbader, E., Valente, T.W.: The stability of centrality measures when networks are sampled. *Social Networks* **25**(4), 283–307 (2003)
5. Freeman, L.C.: Centrality in social networks conceptual clarification. *Social Networks* **1**(3), 215 – 239 (1978)
6. Haveliwala, T.H.: Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *IEEE Trans. on Knowl. and Data Eng.* **15**(4), 784 – 796 (2003)
7. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '03*, pp. 137–146. ACM, New York, NY, USA (2003)
8. Kendall, M.G.: A new measure of rank correlation. *Biometrika* **30**(1-2), 81–93 (1938)
9. Kitsak, M., Gallos, L.K., Havlin, S., Liljerosand, F., Muchnik, L., Stanley, H.E., Makse, H.A.: Identification of influential spreaders in complex networks. *Nature Physics* (2010)
10. Leskovec, J., Faloutsos, C.: Sampling from large graphs. In: *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23, 2006*, pp. 631–636 (2006)
11. Leskovec, J., Kleinberg, J.M., Faloutsos, C.: Graphs over time: densification laws, shrinking diameters and possible explanations. In: *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005*, pp. 177–187 (2005)
12. Li, Q., Zhou, T., Lu, L., Chen, D.: Identifying influential spreaders by weighted LeaderRank. *Physica A: Statistical Mechanics and its Applications* **404**, 47 – 55 (2014)
13. Linyuan, L., Zhang, Y.C., Yeung, C.H., Zhou, T.: Leaders in social networks, the delicious case. *PLoS ONE* **6**(6), 1–9 (2011)

14. Ribeiro, B.F., Towsley, D.F.: Estimating and sampling graphs with multidimensional random walks. In: Proceedings of the 10th ACM SIGCOMM Internet Measurement Conference, IMC 2010, Melbourne, Australia - November 1-3, 2010, pp. 390–403 (2010)
15. Seidman, S.B.: Network structure and minimum degree. *Social Networks* **5**(3), 269 – 287 (1983)
16. Shai, C., Shlomo, H., Scott, K., Yuval, S., Eran, S.: From the Cover: A model of Internet topology using k-shell decomposition. *PNAS* **104**(27), 11,150–11,154 (2007)
17. Valente, T.W., Coronges, K., Lakon, C., Costenbader, E.: How correlated are network centrality measures? *Connections (Toronto, Ont.)* **28**(1), 16–26 (2008)
18. Voudigari, E., Salamanos, N., Papageorgiou, T., Yannakoudakis, E.J.: Rank degree: An efficient algorithm for graph sampling. In: 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), August 18-21, 2016, San Francisco, CA, USA (2016)
19. Zeng, A., Zhang, C.J.: Ranking spreaders by decomposing complex networks. *Physics Letters A* **377**(14), 1031 – 1035 (2013)

Influential Actors Detection Using Attractiveness Model in Social Media Networks

Ziyaad Qasem, Marc Jansen, Tobias Hecking and H.Ulrich Hoppe

Abstract Detection of influential actors in social media such as Twitter or Facebook can play a major role in improving the marketing efficiency, gathering opinions on particular topics, predicting the trends, etc. The current study aspires to extend our formal defined T measure to present a new measure aiming to recognize the actors influence by the strength of attracting new attractors into a networked community. Therefore, we propose a model of an actor influence based on the attractiveness of the actor in relation to the number of other attractors with whom he/she has established connections over time. Using an empirically collected social network for the underlying graph, we have applied the above-mentioned measure of influence in order to determine optimal seeds in a simulation of influence maximization.

1 Introduction

With the wide spread of social media networks nowadays, it has become possible to acquire insights into and knowledge about a wide variety of more or less numerous communities interacting through the Internet. Moreover, applying analytic approaches to social media data can provide better-informed decision-making processes in various fields like marketing, politics, education, etc. In fact, there is an important aspect of such analytics, that is, the detection and characterization of influential actors in social networks. Various studies have suggested different approaches and specific measures to solve the problem of influential actors detection.

Influential actors in social media have an effective role in information diffusion. For instance, A viral marketing operation for a new product can be conducted by

Ziyaad Qasem (e-mail: ziyaad.qasem@hs-ruhrwest.de)✉ · Marc Jansen
Computer Science Institute, University of Applied Science Ruhr West, Bottrop, Germany

Tobias Hecking · H.Ulrich Hoppe
Dept. of Computer Science and Applied Cognitive Science, University of Duisburg-Essen, Duisburg,
Germany

seeding the product in Twitter with a few elected influential actors who can influence others in a way that might help in the rapid spread of that product.

T measure [13] provides a new type of influence in online social network in order to emphasize on those actors who attract many outsiders to join the own community in which a specific topic is dealt. For example, in Twitter those actors spawn many retweets on a certain topic from people who have no previous contributions on that topic.

In this paper, the robust promise of influential actors detection leads us to extend T measure to present a new measure (HT measure) for the detection of influential actors which is based on quantifying the contribution of this actor to increasing the size of the network by attracting new attractors of the specific subcommunity. In other words, while T measure defines the attractiveness value of an actor through evaluating the number of outsiders who joined to the community by this actor, HT measure will refer to his/her attractiveness value through evaluating the importance of those outsiders. In the evaluation section of this paper, we apply our measure to a dataset based on Twitter communication around #EndTaizSiege (related to recent events in Yemen). We compare our measure with T , Katz centrality, indegree, and betweenness measures in terms of how good these measures are if used to refer to the influential actors in social media in terms to their ability to attract others to become active in the information diffusion process.

The rest of the paper is organized as follows: Section 2 presents related research. An overview of T measure approach is given in section 3, which also provides the basic formal definitions. Section 4 introduces the implementation of our measure, followed by the description of our datasets and the experimental results in section 5. Section 6 deals with the performance of our approach in the influence maximization problem. Finally, conclusions are drawn and an outlook for further research is described in section 7.

2 Related Works

Social influence analysis has attracted considerable research interests in recent years. A wide scheme of research focused on modelling and measuring influence and on influential actors detection. Particularly online social networks such as Twitter are of special interest. However, regarding the manifestation and identification there are still open questions.

It could be shown from the study presented by Cha et al. [2] that applying different measures can produce utterly different results when it comes to the task of ranking actors according to their influence. They illustrated an in-depth comparison of three measures of influence: indegree (number of followers of an actor), retweets (number of retweets containing ones actor name) and mentions (number of mentions containing ones actor name). They concluded that different measures can be used to identify different types of influential actors. Popular actors with high indegree were not necessarily influential in terms of spawning retweets or mentions and most influential actors can hold significant influence over a variety of topics. Consequently,

the way in which a network is extracted from social media content and the measure of influence should be considered carefully with respect to the roles and type of influence a one aims to reveal.

Qasem et al. [13] proposed a new approach which is related to the research presented in [2] in the sense that it aimed for a clear formulation of social influence and a methodology to produce an exact ranking of the actors according to the definition. In concrete, Qasem et al. [13] introduced a new type of influence in online social network to define those actors who attract many actors to join the own community in which a specific topic is dealt. Based on this type of influence, a new measure (T measure) has been proposed to define those actors.

In contrast to local measures that only take into account the direct neighbourhood of an actor, there exist also recursive measures that determine the centrality of an actor relative to the influence of its neighbours. A measure of influence proposed in the early years of social network analysis, which is still of importance, is the Katz centrality[7]. It accounts for the ability of an actor to spread information through a network by the counting the number of paths the actors have to each other actor. In addition, longer paths are weighted less than short paths.

Closely related measures are Eigenvector centrality for undirected networks and PageRank for directed networks. These measures are recursive in the sense that they calculate the centrality of each actor based on the centrality of its neighbours. Adaptations to Twitter a based on link analysis are TURank (Twitter User Rank) [16] utilizes ranking algorithm to present based on link analysis a new algorithm in which influential actors are defined. TURank defines actor-tweet graph where nodes are actors and tweets, and links are follow and retweet relationships. PageRank algorithm is extended by TwitterRank [15] to detect influential actors in Twitter based on link structure and topical similarity. Azaza et al. [1] proposed a new influence assessment approach depending on belief theory to combine different types of influence markers on Twitter such as retweets, mentions and replies. They used Twitter dataset of European Election 2014 and deduced the top influential candidates. These ideas were taken up in this work to assess the importance of an actor according to the potential to attract new actors to join the network. Here, the attraction value of an actor can be adjusted by the attraction values of the attracted actors achieve later on. In other words, high attractors are those who influence others to become active in the Twitter communication and also attract many others to do so.

Information diffusion in a network refers often to the influence in the spread of information. Particularly in social media, influential actors can control the diffusion of information through the network to some extent. Information diffusion is defined as the process by which a new knowledge or idea spread over the social networks by the means of communications among the social network actors [14]. The most widely used information diffusion models are the independent cascade (IC) [3][4] and the linear threshold (LT) [5]. These two models describe different aspects of influence diffusion. The IC and LT models have been introduced by Kempe et al. [8] to fix the problem of the influence maximization which search for those actors whose aggregated influence in the social network is maximized. whereas Pei et al. [12] provided strategies to search for spreaders based on the following of information

flow rather than simulating the spreading dynamics (modeled dependent results). Furthermore, The features of identifying spreaders measures using independent interaction and threshold models through empirical diffusion data from LiveJournal are discussed in [11]. Morone et al. [10] proposed to map the problem of influence maximization in complex networks onto optimal percolation using CI (Collective Influence) algorithm.

Our work is related to the research presented in [13] in the sense that we aim to define a new type of influence based on the attractiveness model in order to detect those actors who attract new other attractors to participate the activities of the own community. As well as, our study is related to the approach of [7] in the sense that an actor is influential if he/she is linked from other influential actors. This new type of influence led us to propose a new measure (*HT* measure) to detect those actors, and compare the results with other standard measures. In this paper, we evaluated the performance of our measure in the information diffusion maximization problem by selected sets of top actors based on *HT* measure and other sets which are defined by *T*, Katz measure, and other standard measures.

3 Approach

The approach of *T* measure provides a new type of influence in online social network in order to emphasize on those actors who attract many outsiders to join the own community in which a specific topic is dealt. Thus, influential actors who are detected by *T* measure are those actors whose tweets spawn many retweets in a way that leads to an increase in the size of social network. *T* measure depends on the decomposition of a topical dataset that is collected from a social network according to the time period of collection. The basic idea of the dataset decomposition is to analyze a specific event in social media after each slice of time. The aim is to define the actors who affect the size of this event by attracting outsiders to participate. To be more specific, the attractiveness value (*T* value) of the actor *A* in the slice time *t* equals the number of new attractors who joined the community in the slice time *t* + 1 by establishing new connection with actor *A*.

To formalize our *HT* measure, we will enumerate here briefly some of the concepts that are used to implement *T* measure.

The approach of *T* measure is based mainly on the decomposition of a topical dataset that is collected from a social network according to the time period of collection. This time period is referred to by the term *P*-period.

Definition 3.1 (*P*-period). *P*-period is a time duration of the data collection process from social networks.

The definition above is applied to the streaming dataset obtained from online social networks. If we have a historical dataset, *P*-period will be the period between the oldest activity (in Twitter, the activity would be tweet, retweet, reply, etc.) and the newest one in that dataset.

The social networks dataset in this approach is represented by a directed graph which is referred to by P -graph.

Definition 3.2 (P -graph). P -graph is a directed graph constructed from social network data which have been collected during P -period.

Decomposition of a P -graph leads to decomposition of the P -period into slices of time so that every subgraph is related to a slice. This slice is referred by P -slice.

Definition 3.3 (P -slice). P -slice is a time slice of P -period.

If all P -slices are equidistant, the P -slice is called EP -slice.

Definition 3.4 (EP -slice). EP -slice is a P -slice in case all P -slices are equidistant.

To ease the definition of subgraphs of this approach, some terms related to actors according to P -slices are defined.

Definition 3.5 (P -actors). Let s_1, s_2, \dots, s_n be the P -slices. For every i such that $0 < i \leq n$, the P -actors A_i is a set of all actors that joined the social network between the P -slices 0 and s_i .

Definition 3.6 (P_s -actors). Let s_1, s_2, \dots, s_n be the P -slices. For every i such that $0 < i \leq n$, the P_s -actors A_{s_i} is a set of all actors that joined the social network between the P -slices s_{i-1} and s_i .

Figure 1 shows how the P -actors and P_s -actors are taken with respect to P -slice in this approach. The figure displays the P -actors A_3 and P_s -actors A_{s_3} as an example. A_3 is the set of all actors who joined the community until s_3 whereas A_{s_3} joined between P -slices s_2 and s_3 .

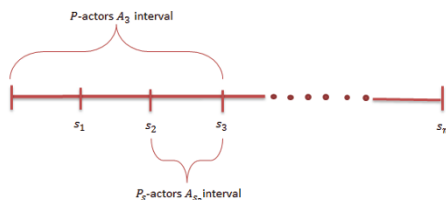


Fig. 1: P -actors and P_s -actors with respect to P -slices

The subgraphs used in this approach are defined as the following:

Definition 3.7 (P -subgraph). P -subgraph $G_i(A_i, E_i)$ is a directed subgraph of P -graph which is aggregated until P -slice s_i .

Definition 3.8 (S -subgraph). The i -th directed S -subgraph $S_i(A_i, E_{s_i})$ is the subgraph of the directed P -subgraph $G_i(A_i, E_i)$ with $E_{s_i} = \{(a, b) : (a, b \in A_{s_i}) \text{ or } (b \in A_{i-1} \text{ and } a \in A_{s_i})\} \cap E_i$

Figure 2 shows the difference between P -subgraph and S -subgraph in this approach where n is the number of P -slices and $1 < i \leq n$. P -subgraph G_{i-1} is the P -subgraph of the P -slice s_{i-1} , and P -subgraph G_i and S -subgraph S_i are of the P -slice s_i .

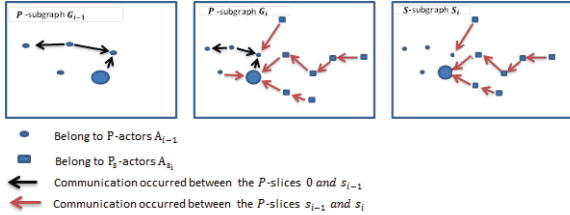


Fig. 2: Directed P -subgraphs G_{i-1} and G_i , and directed S -subgraph S_i

In the next section, we will introduce the implementation of our measure based on this approach.

4 Implementation

T measure tries to define those actors who attract many actors to the community. Figure 3 shows how the attractiveness value of the actor A is calculated with respect to T measure.

From figure Figure 3, T value of the actor A in the P -subgraph $G_{(i-1)}$ is equal to its indegree value in the S -subgraph S_i . Hence, The number of new actors joined the community by the actor A .

$$T(A_{G_{i-1}}) = indegree(A_{S_i}) \tag{1}$$

The indegree measure evaluates the number of neighbors of the actor A with order 1 (number of the immediate neighbors). In HT measure, we will increase the order to include the neighbors with order m , where m is the maximum neighborhood order. Thus, HT measure defines the attractors of attractors. Figure 4 shows the difference between T measure and HT measure.

From figure 4, HT value of the actor A in the P -subgraph $G_{(i-1)}$ is equal to its indegree plus the indegree of his/her neighbors with order m in the S -subgraph S_i .

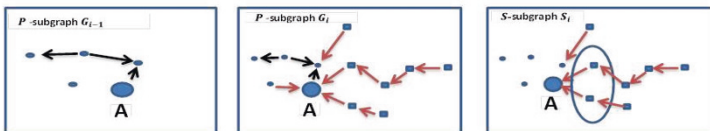


Fig. 3: T measure evaluation

$$HT(A_{G_{i-1}}) = T(A_{G_{i-1}}) + \sum_{a \in neighbors(A_{S_i}, m)} indegree(a_{S_i}) \tag{2}$$

Where m is the maximum neighborhood order.

HT and T values of the actor A in whole P -graph G are calculated as following:

$$T(A_G) = \sum_{i=1}^{n-1} T(A_{G_i}) \tag{3}$$

$$HT(A_G) = \sum_1^{n-1} HT(A_{G_i}) \tag{4}$$

Where n is the number of slices.

5 Evaluation

In this section, we will describe the evaluation strategy. Furthermore, the experimental results on the dataset will be discussed in this section.

5.1 Evaluation Strategy

We gathered a dataset from Twitter via Twitter API from December 31, 2015, to January 06, 2016. This Twitter dataset relates to the hashtag #EndTaizSiege (14,944 actors and 46,552 connections) that comprises a big connected component (containing 84% of actors), singletons (14%), and smaller components (2%).

Applying our approach leads to decompose P -graph constructed from Twitter dataset into three P -subgraphs and two S -subgraphs based on three P -slices. As a matter of fact, the time slicing has been estimated in accordance to the size of dataset using an equal window size for each slice. Figure 5 shows how the P -period with Twitter dataset #EndTaizSiege has been decomposed into equal window size so that we get a fair division of the retweet activities for each time slice.

The directed weighted P -graph of our collected Twitter dataset is constructed based on retweet activities so that actor a gets incoming connection from actor b if actor b retweeted a tweet of actor a . The weight of connection refers to the number of retweets between two connected actors.

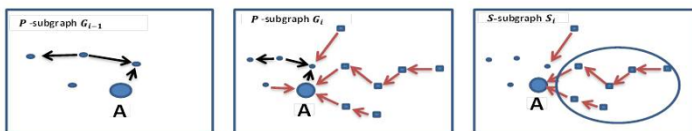


Fig. 4: HT measure evaluation

5.2 Experimental Results

For our Twitter dataset, we applied *HT* measure to verify whether it can detect influential actors. Table 1 shows the description of the top influential actors with respect to *HT*, *T*, Katz centrality, indegree, and betweenness measures. The question mark in the table 1 fields refers to an actor who is not a well-known as an influential actor within the community. We notice here how the *HT* and *T* measures refer to well-known influential actors within the community, or to the famous news accounts. Unlike other measures, the top ten influential actors with respect to *HT* and *T* measures are well-known within the community. In our case, the well-known actors have been recognized based on a local expertise, where they are the most renowned actors in the field of human rights and politics who continually traded their names in the newspapers and news concerning the current situation in Taiz city in Yemen. Their names have not been mentioned explicitly in order to protect their privacy.

Table 1: Description of top influential actors according to different influence measures in Twitter dataset #EndTaizSiege

Rank	HT	T	Indegree	Betweenness	Katz Centrality
1	News Account N1	News Account N1	News Account N1	?	News Account N1
2	TV announcer T1	Journalist J1	Journalist J1	?	?
3	Journalist J1	TV announcer T1	TV announcer T1	?	Human Rights Activist H1
4	Human Rights Activist H1	Television reporter R1	Journalist R3	Journalist J2	Journalist J2
5	Human Rights Activist H2	Human Rights Activist H1	Human Rights Activist H1	?	?
6	Television reporter R1	Human Rights Activist H2	News Account N2	?	Television reporter R1
7	News Account N2	News Account N2	Human Rights Activist H2	Human Rights Activist H3	Journalist J1
8	Journalist J2	Political activist P1	?	TV announcer T1	TV announcer T1
9	Political activist P1	Journalist J2	Political activist P1	News Account N1	?
10	Political activist P2	Political activist P2	?	?	?

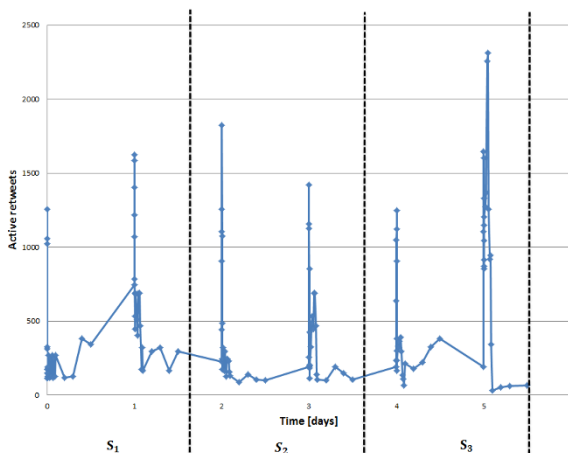


Fig. 5: Retweet activities over time in our Twitter dataset

6 Information Diffusion

In our work, we study the information diffusion to compare our measure with other existing measures in terms of how good these measures are if used to refer to the influential actors in social media in terms to their ability to attract others to become active in the information diffusion process. In order to assess how well the *HT* measure is suited to uncover influential actors with respect to information diffusion, we simulate the diffusion of information originating from a seed set of nodes through the Twitter networks using the well-known independent cascade (IC) model [8].

In information diffusion, the IC model is proposed where the information flows through cascade over the social network. In IC model, there are two terms are used to describe the state of the actors. The actor who is influenced by the information is called active, and inactive for the actor who is not influenced. The IC model process starts with activated actors as an initial seed set . In step s , an actor a will get a single chance to activate each currently inactive neighbor b . Actually, the activation process is based on the propagation probability P of the actors links. The propagation probability P of a link is the probability by which an actor can influence the other actors. In Twitter, we proposed that actor a is influenced by actor b if he/she retweeted from actor b in proportion to the tweets number of actor b . So, the propagation probability P in IC model is based in our Twitter dataset on the link weight divided by tweets number of target actor.

To compare the performance of actors sets selected by the *HT* measure with other influence measures, we selected sets of top actors based on the *HT*, *T*, and Katz centrality measures. As well as, we selected the sets identified by measures that are known to be good heuristics for seed set selection, namely degree and betweenness centrality [9].

6.1 Simulation of attraction processes with time-respecting paths

In this section, We will report results based on simulated attraction processes. To do so, we adapt the IC model that is known to simulate the diffusion of information through a network as described above. Information diffusion and attraction processes have some commonalities but differ on various aspects. In traditional information diffusion models such as the IC model, the network is usually considered as stable in the sense that the set of nodes and the set of edges do not change over time. However, the nodes changes their states "inactive" and "active" during the information diffusion process. Attraction, as it is studied in this paper is similar in the sense that actors who are not part of the community (i.e. do not have contributed a tweet) are inactive while others are considered as active. On the other hand, the original IC model does not account for the fact that the network grows when new actors become attracted to the community. Thus, the IC model was adapted to take into account the creation times of the edges. These time varying networks have special characteristics regarding reachability of node pairs since a walk on the graph can only take edges with increasing timestamp, which is known as the time-respecting property (see [6]). In this aspect, we added a new activation rule to the IC model which is: the actor who is activated in time t

cannot activate those actors who have been linked with him/her before the time t . To explain this activation rule in more details, we define the following terms:

Definition 6.1 (Path-time). The path-time of each link in the network is the P -slice number in which this link has been created.

Definition 6.2 (Activation-time). The activation-time of each activated actor is the path-time of the link by which this actor has been activated.

Now, we can state that the actor a can not activate the actor b if the link from b to a has a path-time later than the activation-time of the actor a .

Using this activation rule the simulation can be interpreted as an attraction process where actors who are already part of the communities can attract others only if their activity starts after the activator has become active.

Previous studies [13] have shown that a seed selection strategy based on indegree yields similar results as a selection strategy based on the T measure. This is also expected with respect to the high correlation between these two measures. However, the benefit of the T measure that distinguishes it from other measures is that time is explicitly taken into account. The experimental results in the next section support the assumption that the T and HT measure can identify important attractors in time varying networks while it boils down to indegree if time is neglected.

6.2 Experimental results

Here, we considered the dataset of #EndTaizSiege which is related to an organized event in Yemen. Hence, we got a highly connected component that is suitable for the application of our approach which is basically aimed to identify those actors who contribute to attract others to participate in a specific organized event. We simulated the information diffusion based on the IC model with time-respecting paths for seed sets of sizes $n = 1...25$ which are generated from different influence measures. The diagram in figure 6 shows the results of applying IC model on our Twitter dataset with different seed sets which identified by different influence measures. Comparing with other influence measure, we notice that the HT measure yield the best performance in information diffusion under the IC model with time-respecting paths for the seed sizes bigger than 11. Additionally, we statistically verified the results of simulation for each seed set using T-Test. In case of $n > 11$, the differences between HT and T measures are significant. For example, results for the seed set 12 show that there is a significant difference in the score of HT measure ($M = 1259.95$; $SD = 291.1128$ conditions; $t(19) = 3.678480757$; $P = 0.000$). On the other hand, the differences among HT and indegree measures are also significant in case of $n > 12$.

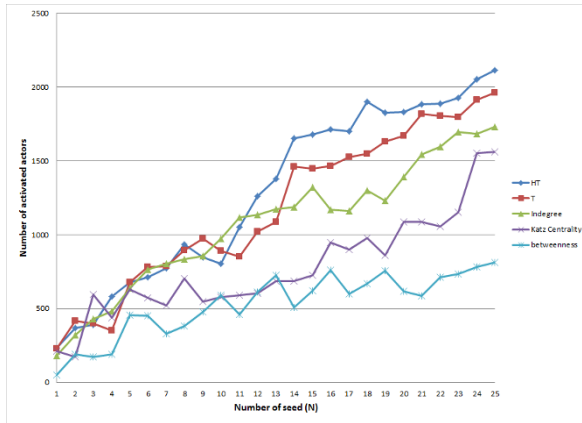


Fig. 6: IC model under time-respecting paths with different influence measures over Twitter dataset #EndTaizSiege

7 Conclusion

In summary, we presented in this paper an extended approach to detect influential actors based on the attractiveness model that is introduced with T measure. Our approach detects those actors who contribute effectively to increase the size of social network by attracting new attractors to the community in which a specific topic is dealt. Through experiment results we presented through how our proposed measure HT referred to the influential actors in Twitter dataset. Furthermore, we showed through experiment and statistical tests that the best performance has been yielded by HT measure in maximization of influence problem when we took the time into account.

Our current work in extending and improving this approach focuses on an elaboration of our measure with more datasets and more results, and describe it on multi-layer networks. Furthermore, we plan to develop an efficient general strategy for time slicing to determine the time period decomposition into time slices, and the role of time slicing in making HT measure far better than existing measures.

References

- [1] Azaza, L., Kirkizov, S., Savonnet, M., Eric, L., Faiz, R.: Influence assessment in twitter multi-relational network. In: Proceedings of the 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), pp. 436–443 (2015)
- [2] Cha, M., Haddadi, H., Benevenuto, F., Gummadi, P.K.: Measuring user influence in twitter: The million follower fallacy. ICWSM **10**, 10–17 (2010)
- [3] Goldenberg, J., Libai, B., Muller, E.: Talk of the network: A complex systems look at the underlying process of word-of-mouth. Marketing letters **12**, 211–223 (2001)
- [4] Goldenberg, J., Libai, B., Muller, E.: Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic

- cellular automata. *Academy of Marketing Science Review* **9**, 1–18 (2001)
- [5] Granovetter, M.: Threshold models of collective behavior. *American journal of sociology* pp. 1420–1443 (1978)
 - [6] Holme, P., Saramäki, J.: Temporal networks. *Physics reports* **519**, 97–125 (2012)
 - [7] Katz, L.: A new status index derived from sociometric analysis. *Psychometrika* **18**(1), 39–43 (1953)
 - [8] Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: *Proceedings of the 9th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146 (2003)
 - [9] Mochalova, A., Nanopoulos, A.: On the role of centrality in information diffusion in social networks. In: *ECIS* (2013)
 - [10] Morone, F., Makse, H.A.: Influence maximization in complex networks through optimal percolation. *Nature* (2015)
 - [11] Pei, S., Makse, H.A.: Spreading dynamics in complex networks. *Journal of Statistical Mechanics: Theory and Experiment* (2013)
 - [12] Pei, S., Muchnik, L., Andrade Jr, J.S., Zheng, Z., Makse, H.A.: Searching for superspreaders of information in real-world social media. *Scientific reports* (2014)
 - [13] Qasem, Z., Jansen, M., Hecking, T., Hoppe, H.U.: On the detection of influential actors in social media. In: *Proceedings of the 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pp. 421–427 (2015)
 - [14] Rogers, E.: *Diffusion of Innovations*, 5th Edition. Free Press, New York (2003)
 - [15] Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: *Proceedings of the third ACM international conference on Web search and data mining*, pp. 261–270 (2010)
 - [16] Yamaguchi, Y., Takahashi, T., Amagasa, T., Kitagawa, H.: Turank: Twitter user ranking based on user-tweet graph analysis. In: *Web Information Systems Engineering–WISE 2010*, pp. 240–253. Springer, Berlin Heidelberg (2010)

Analyzing Multiple Rankings of Influential Nodes in Multiplex Networks

Sude Tavassoli and Katharina A. Zweig

Abstract In many networks, different centrality indices reveal conflicting rankings of the nodes. The problem is worsened, if the same nodes occur in different but related network layers, i.e., in multiplex networks. The main concern in the analysis of multiplex networks is maintaining the inherent nature of multiple layers in the explorations. Therefore, in this paper we discuss a method combining a fuzzy operator with a visualization, that allows the exploration of a node's centrality with respect to different network processes on different layers of the same network simultaneously. Our empirical results indicate that an airport transportation network allows for a smaller number of different behaviors than social networks in a medium sized law firm and a large sized tweet dataset.

1 Introduction

Freeman already pointed out in 1978, that the concept of centrality can be characterized in different ways using a number of centrality indices [8]. The wide range of proposed centrality measures confirms the success of this simple concept in the analysis of static network structure [4, 9, 11, 13, 16]. However, for a long time, there was neither a full, non-trivial characterization containing all known centrality indices nor a theory that explained when to use which of the dozens of centrality indices [14]. Finally, in 2005, Borgatti stated that centrality indices predict the importance of a node with respect to a process on a given infrastructure [3], e.g., spreading a rumor [5] or propagating an infectious disease [10]. Thus, for any single process, only one centrality index fits, according to Borgatti [3]. The question that arises is what happens if multiple processes take place in a network and if one wants to analyze the centrality of the given network? In our previous work, we proposed to use a fuzzy

Sude Tavassoli (e-mail: tavassoli@cs.uni-kl.de)✉ · Katharina A. Zweig (e-mail: zweig@cs.uni-kl.de)
Computer Science Department, Kaiserslautern University of Technology, 67663 Kaiserslautern, Germany

operator based on *at least one, a few, almost all, or all* the processes in that set [17]. It has been elaborated in a study that the complicated nature of complex systems entails going beyond the analysis of single-layer networks and considering multi-layer or multiplex networks where agents interact using multiple types of relations or interactions [12]. Therefore, a wide range of recent studies suggested methods to analyze these networks, such as structural measures [2], link assessment [1], and centrality ranking in multi-layer networks [16]. Similarly, the question that arises is how to deal with a node's centrality index in multiple layers, or—worse— multiple centrality indices of a node in multiple layers. First approaches simply aggregated the result of centrality indices over the layers, e.g., by averaging over all indices in all layers. However, the aggregation of the classical centrality indices can result in misleading results [6, 16] and suppresses possibly interesting information.

In our recent studies [17, 18], we considered the evaluation of nodes' centrality as a Multi Criteria Decision Making (MCDM) problem. In that setting, several normalized centrality indices play the role of multiple criteria and nodes were considered as alternatives; if a node gets a high normalized index of centrality, it obtains a high satisfaction value of the corresponding criteria. The best solution among the alternatives can be selected with respect to the chosen multiple criteria which are, e.g., the normalized classical centrality indices: Degree, Betweenness, Closeness, and Eigenvector. Likewise in this paper, we analyze the influence or importance of nodes with respect to multiple centrality indices but in multiplex networks.

1.1 Research questions

Considering multiple aspects of centrality based on a set of network processes of interest, brings up the question of how conveniently the influence of a node can be analyzed with respect to *at least one* process, *most* of them or *all* of them within a layer and over multiple layers. In most studies, a regular average over multiple centrality indices and/or over the layers is employed. Building the average is convenient, but it has been suggested that it is not an ideal option when multi-layer, interconnected, or multiplex networks are concerned [6, 16]. In addition, it misses the information whether one node is especially important for at least one of the network processes, or whether there is a node that is never very influential, but at least moderately influential for all network processes.

Thus, our research questions are the following: (1) Do rankings based on a set of centrality indices rather correlate or conflict? (2) If they conflict, how can the different aspects of centrality be explored for each node within one layer? (3) How can the different aspects of centrality be explored for each node within all layers of interest? (4) How can the patterns of centrality rankings of nodes be analysed within all layers? We show how one kind of visualization can help to understand conflicting centrality indices rankings. For this, we consider the normalized centrality indices themselves as multiple criteria in a decision making problem. We then use a fuzzy operator to find the best solution, i.e. the most influential node, based on the possibly conflicting centrality indices. Finally, we propose two measures that allow

us to partition the nodes into the different groups and to explore the nodes that have a similar pattern of centrality rankings.

2 Definitions, data, and methods

A multiplex network is defined as follows: it is a network with $|M|$ layers $M = \{l_1, l_2, \dots, l_{|M|}\}$ where each layer l_i itself is a network comprised of $|N_i|$ nodes and $|E_i|$ edges. Each edge set E_i represents a different type of relation or interaction, and in almost all multiplex networks some nodes are contained in multiple layers. Let $d_i(v, w)$ denote the distance of two nodes in layer l_i which is defined if and only if $v, w \in N_i$. The degree $deg_i(v)$ is defined as the number of edges it is contained in layer l_i . The closeness centrality $close_i(v)$ of a node is defined as the sum of all distances of v to all other nodes in N_i . The betweenness centrality $betw_i(v)$ is defined as $\sum_{s,t \in N_i} \frac{\delta_{s,t}(v)}{\delta_{s,t}}$, where $\delta_{s,t}(v)$ denotes the number of shortest paths between s and t that contain v and $\delta_{s,t}$ denotes the number of all of their shortest paths.

2.1 Data sets

We use the following three multiplex network data sets:

1. The **Europe Airlines dataset** is a multiplex network dataset which has been developed by Cardillo et al. (2013) [4]. The dataset contains an undirected and unweighted network comprised of 37 layers where each layer corresponds to an airline in Europe, including high cost airlines (Lufthansa, British airways, and Air France) and low cost airlines (Airberlin, Ryanair, and Easyjet). Each node represents an airport and two nodes are connected if there is at least one direct flight between them. For the experiments in this paper, we use the three layers of low cost airlines, which share 20 airports.
2. **Law firm data set** is a 3-layer multiplex network provided by Lazega (2001) [15] in the study of how 71 attorneys of a law firm go forward on the same task based on their social ties which namely represent seeking advice (directed relationship), co-working, and friendship. In the first layer, a node is connected to the other nodes to whom he/she might go for taking advice on a task. The second layer of network contains the ties between two nodes if they are co-workers. Note that, the advisor is not necessarily a co-worker or wise versa. In the third layer, the nodes are connected if they socialized outside the firm.
3. A *tweet network* called the **Higgs Boson dataset**, compiled by De Domenico et al. (2013) [5], includes four directional network datasets. The nodes are the users and there is a directed edge between a pair of nodes in the first three networks, if one user replied to another one, retweeted the post, or mentioned the other user in his/her tweet about the Higgs particle. The fourth network contains the social interactions of the nodes for being friends/followers. Our analysis is restricted to the biggest, strongly connected component of each of the first three networks, which have 127 nodes in common.

2.2 Identification of influential nodes as an MCDM

As discussed earlier, a node might be considered central for some network processes but not for all, even within a single layer. This problem becomes even more complex, if the centrality of multiple layers is concerned. We consider the analysis of the different normalized centrality indices (degree, betweenness, and closeness) of a node within one layer as an MCDM problem. An MCDM tries to find a satisfying solution among alternatives with respect to multiple, possibly conflicting criteria—as is the case for most centrality indices that almost never agree perfectly on the ranking. The nodes are considered as the alternatives in this decision making where the best solution (the most influential node) can be selected based on the satisfaction of either *at least one* criterion, *most*, or *all* of them or anything in between. Fuzzy operators provide a means to scale between these extremes in a seamless way, guided by some parameter.

Maximum Entropy Ordered Weighted Averaging is one of the fuzzy operators proposed by Yager to solve an MCDM problem [7, 19, 20]. He assumes that the extent to which a criterion is met is expressed by a value between 0 (no satisfaction) and 1 (full satisfaction) and considered various ways of aggregating these possibly conflicting values into a single result, which can then be used to rank all alternatives. He stated that the aggregation of multiple criteria in a decision making problem for a solution can be scaled between two extreme cases of pure *OR* and pure *AND*. In the pure *OR*, the maximum value of satisfaction obtained from **any** criteria has the most important role in the aggregation. In the pure *AND*, the role of the minimum value of satisfaction among the criteria determines the aggregation. The *OR* operator thus represents the situation in which *at least one* criterion with the best satisfaction value is enough to give an alternative the highest rank and the *AND* operator represents a situation in which *all* the criteria needs to be satisfied to result in a high rank. Yager showed that anything between these two extreme cases can be represented using proportional linguistic quantifiers such as *a few*, *most*, and *almost*, as introduced by Zadeh [21]. For each alternative x , MEOWA operator uses $A(x)$, the vector of its n satisfaction values, where all values are between 0 and 1. Then, these values are sorted non-increasingly in vector $B(x)$. Note that the order of the criteria is in general different for each of the alternatives! The aggregation is then computed as the scalar-product of a weight vector W and $B(x)$:

$$\lambda(a_1, a_2, \dots, a_n) = \sum_j W_j \cdot B(x)_j$$

The weight vector itself is obtained using the following function based on some parameter β [7]:

$$w_i = \frac{e^{\beta \frac{n-i}{n-1}}}{\sum_{j=1}^n e^{\beta \frac{n-j}{n-1}}}$$

The resulting weights are always between $[0, 1]$ and their sum is equal to 1. It can be easily seen that high values of β lead to a weight vector that gives a weight close to 1 to the **first** position of the sorted vector $B(x)_j$, i.e., the result is dominated by the maximum satisfaction value. This is considered to be a high *orness* - it is enough if one criterion is strongly satisfied. A high, negative value of β favors the last position

in the sorted vector $B(x)_j$, i.e., the **least** value. This is considered a high *andness*. Note that for $\beta = 0$, the weight vector contains $1/n$ in all positions, i.e., an average of the satisfaction values is computed. For all values of β , an *orness* measure denoted by Ω is defined by Yager [7]:

$$\Omega = \frac{1}{n-1} \sum_{i=1}^n (n-i) \frac{e^{\beta \frac{n-i}{n-1}}}{\sum_{j=1}^n e^{\beta \frac{n-j}{n-1}}}$$

For the β -values of -20 and 20 , the *orness* equals 0 and 1 respectively. The *orness* is 0.5 for $\beta = 0$.

3 Experimental Results

3.1 Air-transportation network

Via the air-transportation network, different centrality indices are of interest: a direct property indicating importance is the number of flights reaching a city, as measured by the degree—it can be assumed that it correlates with the number of people wanting to go there (by a specific airline). Another indicator of importance is the average distance to an airport which is directly proportional to its *closeness*. It might be interpreted as the ease by which an infecting disease reaches this airport. Finally, the *betweenness centrality* is associated with a network process that uses shortest paths; it is directly proportional to the average fraction of shortest paths that would be lost if that airport was shut down, between any two airports taken at random. For twenty airports shared between all three layers of low cost airlines, these three centrality indices were measured in each layer and normalized by the maximum and minimum observed values for the corresponding index.

The first question to be addressed is that whether the rankings regarding the chosen centrality indices actually conflict or whether they correlate strongly. Figure 1a shows a pairwise scatter plot of two of the chosen centrality indices. While there is a general positive correlation, there are always conflicting views on the same node. Thus, an analysis with a fuzzy operator is meaningful and can be used to explore these conflicts in a convenient manner. Figure 2a shows, for each of the shared airports and each of the three low-cost airlines *Airberlin*, *Ryanair*, and *Easyjet*, the airports' ranking position within each of the layers for different values of β . By concentrating on all curves of the same color, a comparison of *within-layer influence* regarding the three chosen network processes is possible, as shown in the following. In the layer of Airberlin, it can be seen that the airports of Palma de Mallorca and Kos Island obtain the highest and the second highest rank among the airports within the layer, independent of β . Faro airport ($[0.48, 0.107, \mathbf{0.72}]$) is also an airport with an almost stable ranking position, but there are always nodes with even higher values. In the high *orness* (right side of the plot), for example, it is located lower than the airport of Alicante with the normalized centrality values of $[0.44, 0.098, \mathbf{0.732}]$, because Alicante's last value is a tad higher than Faro's last value. But to the left side of the subplots (high *andness*), the ranking of Alicante is demoted, since its smallest

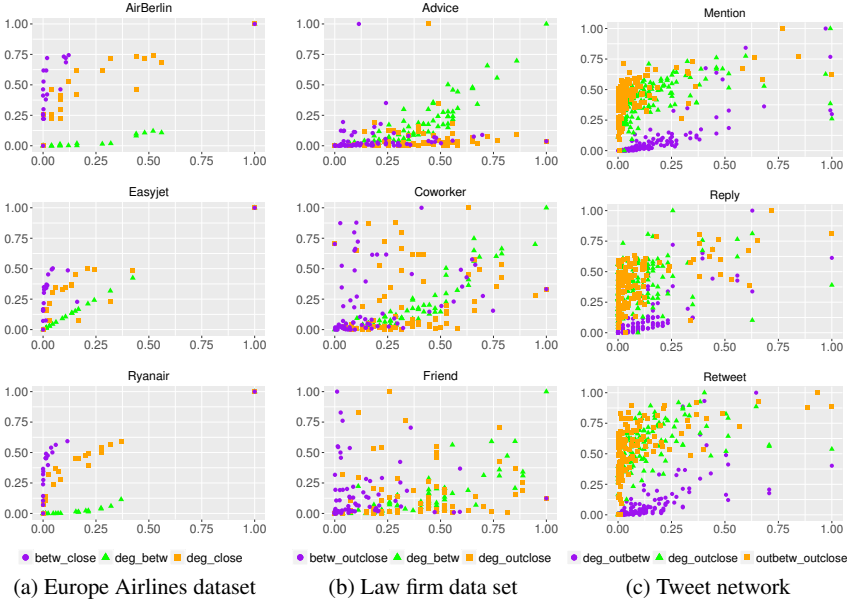


Fig. 1: The correlations between the three normalized centrality indices are depicted for each layer of three multiplex networks respectively.

value of satisfaction (0.098) is less than that for the Faro airport (0.107). In the layer of Easyjet, the airport of Gatwick always occupies the highest rank, independent of the β -value, i.e., its ranking pattern is similar to London airport in the layer of Ryanair. As mentioned, in the layer of Ryanair, not very surprisingly, London is first with respect to all chosen network processes, while, maybe more surprisingly, the airports of Alicante and Madrid are always second and third. We can also use the same visualization to understand the influence of one node (airport) with respect to all three airlines and all three network processes of interest. The very first observation is that there is no airport that is most influential in all three layers at the same β —it seems that the low-cost airlines rather partition the market than share it. However, the airports of Málaga and Alicante are always among the top 6 influential nodes in all three layers.

In order to address the fourth research question, we use ΔAgg , which measures the maximum difference in ranking positions fixing a layer and ΔLayers , which measures the maximum differences in ranking positions fixing a β -value. First, we obtain the minimal rank of node v within layer L_i over all β -values and denote it by $\min\text{Rank}(v, L_i)$ and obtain $\max\text{Rank}(v, L_i)$ accordingly. Then, $\Delta\text{agg}(v) := \max\{\max\text{Rank}(v, L_i) - \min\text{Rank}(v, L_i) \mid 1 \leq i \leq |L|\}$; a large value of ΔAgg means the centrality indices where more conflicting. Note that, the $\max\text{Rank}(v, L_i)$ can be found in a β -value in the range of $[-20, 0)$ or in $[0, 20]$. For the categorization, we count the number of times that the $\max\text{Rank}$ among $|L|$ layers is obtained in a β -value in

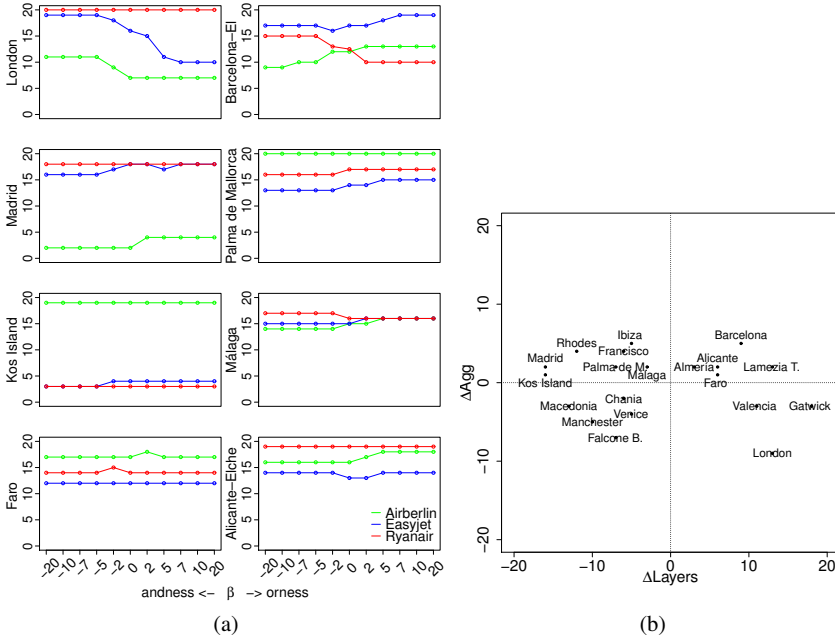


Fig. 2: (a) Rankings of some airports shared between the three layers of airlines using the different values of β . (b) Categorizing of the shared nodes (20 airports in total) using two proposed measures of ΔAgg and $\Delta Layers$.

$[-20, 0)$. If the measured frequency ($FmaxRank(v) \geq k$), then $-\Delta agg$ is assigned to the node v , otherwise ΔAgg ; this partitions the nodes into two groups. In the first group, the nodes' least centrality value among three indices is high enough to give them a high rank in the high *andness* and in the second group—above the horizontal line—the nodes' maximal centrality index value is high enough to prioritize them in the high *orness*.

The maximal differences among all layers for node v for any β -value can be measured using $maxRank(v, \beta)$, which is the maximal rank of v based on any layer and $minRank(v, \beta)$ is defined as minimal rank for any β -value. The overall maximum differences of node v is then defined as $\Delta Layers(v) := \max\{maxRank(v, \beta) - minRank(v, \beta) | \beta \in \Gamma\}$, where Γ is a set of β -values. A large value of $\Delta Layers$ indicates, the node v is more influential in one or two layers and not influential in the rest. In the categorization, if the max value has been obtained in a β -value in $[-20, 0)$, then $-\Delta Layers$ is assigned to node v , otherwise $\Delta Layers$; this again partitions the nodes using a vertical line into two groups. We choose $k = 2$ as the number of layers in the used multiplex networks is only three.

As shown in Figure 2b, for instance, Madrid airport obtains $\Delta agg = 2$, which indicates that this airport has almost stable ranking fixing one layer using different aggregation strategies and its *maxRank*-values have been found in at least two layers

towards the high *orness*. Instead, it has a high difference of ranking among all layers ($\Delta Layers = -16$), i.e., very central in two layers and not central in the rest. Its maximum difference has been found in a β value toward the high *andness*. The interesting point of this visualization is that we observe the nodes that have similar ranking patterns considering multiple layers. For example, Madrid and Kos Island have similar patterns considering both ΔAgg and $\Delta Layers$. London and Barcelona are located in two different groups. London often obtains the maximum rank in the high *andness* and in contrast, Barcelona achieves it in two layers in the high *orness*, but, they both obtain positive $\Delta Layers$ -values. Madrid and London airports are exactly located in opposite groups.

3.2 Law firm data set

In the law firm data set, one important network process is again the direct influence someone might have on other people, as quantified by the degree centrality. Regarding communication flows in small groups, the betweenness centrality might again reflect the influence of a person. Since we also have directed relations in this data set, the last network process of interest is the average minimal number of steps to give a message to another person—as quantified by the *out-closeness*, the analogous, directed version of the classic closeness. Figure 1b shows that the different centrality indices have very different ideas about who is most influential with respect to the network process they represent.

In the layer of Advice as shown in the Figure 3a, node 1 is among the three most influential nodes with respect to all three normalized indices of $[0.442, 0.114, \mathbf{1}]$ —it is also interesting to see that the degree, the number of people seeking advice from her or him, is not maximal. It achieves a maximal value in the out-closeness. The other top ranks in this layer are node 26 in the highest place with the indices of $[\mathbf{1}, \mathbf{1}, 0.037]$ (maximal betweenness) and node 24 in the fourth place: $[\mathbf{0.767}, 0.557, 0.042]$, also based on a high degree. Note that the node's lowest satisfaction value at the out-closeness is really very small. This gives node 24 a medium to high rank when the ranking considers the node's influence with respect to *all* network processes of interest. In the layer of Coworker where a lawyer (as a node) is connected to the other nodes if he/she spent time with them on a law case. Interestingly, node 24 and 4 which we already analyzed in the Advice layer, are among the top 2 in the high *orness* with respect to their normalized centrality indices of $[\mathbf{1}, \mathbf{1}, 0.332]$ and $[0.632, 0.41, \mathbf{1}]$, respectively. Another interesting case is node 3 which is one of the nodes with a sharp decreasing from the high *orness* to the high *andness*. It turns out that this node has the least number of coworkers but in terms of being indirectly close to other coworkers of coworkers, he/she obtains a much larger value $[0, 0, 0.703]$. Thus, when at least one criterion is enough, the corresponding lawyer is one of the top 10 influential persons in the law firm, but both, on average and when all network processes are considered, this node gets the least ranking position. As can be seen in Figure 3b, the number of nodes with almost similar ranking patterns as node 3 is not small in the top right category. In the layer representing friendship, in the high *orness*,

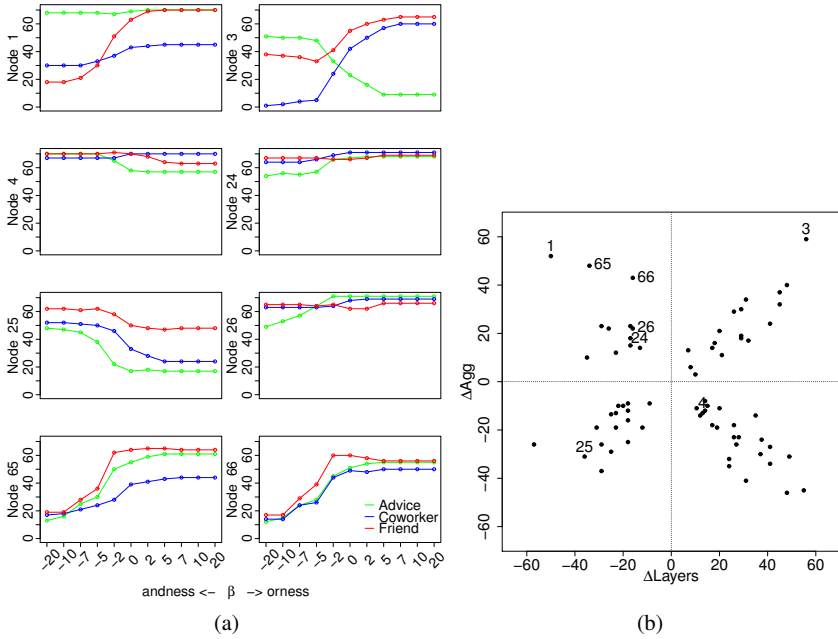


Fig. 3: (a) Rankings obtained using the different values of β -parameter for some selected nodes out of 71 nodes in all three layers of relations. (b) Categorizing of the 71 nodes using two proposed measures of ΔAgg and $\Delta Layers$.

nodes 1 and 24 are top two nodes with the normalized indices of $[0.259, 0.011, 1]$ (maximal out-closeness), and $[0.889, 0.341, 0.186]$, respectively. However, node 1 is one of the nodes in this layer that has one of the smallest minimal satisfaction values and thus its rank drops significantly for the high *andness*.

3.3 Tweet network data set

A tweet network, especially of a very large size, definitely supports direct influence as measured by the degree, but in our view it is not likely to support any network process that uses shortest paths and assumes that all pairs of nodes want to communicate with each other or learn of each others' interest with the same frequency. However, the closeness and betweenness centrality indices assume exactly this: equal need of communication along shortest path between all pairs of nodes. However, for consistency with the other data set and as a pure demonstration, we stick to the normalized indices of degree, out-betweenness and out-closeness. Again, these centralities do not correlate very strongly (s. Figure. 1c).

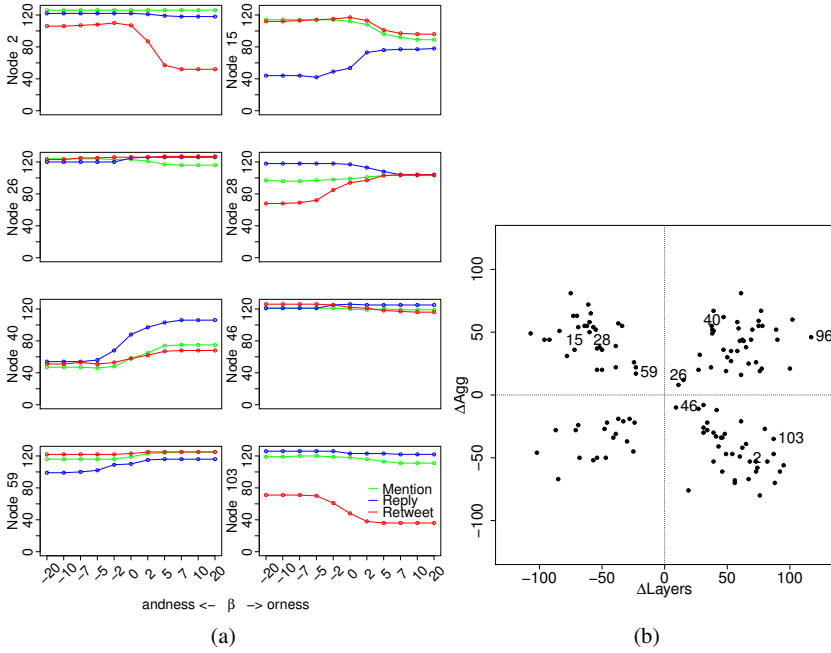


Fig. 4: (a) Rankings obtained using the different values of β -parameter for some shared nodes between the three layers of the Higgs Boson dataset. (b) Categorizing of the 127 shared nodes using two measures of ΔAgg and $\Delta Layers$.

We use the degree of these 127 nodes obtained in the fourth layer as an additional information for the exploratory analysis, i.e., the number of their friends/followers on Twitter. This additional information allows for another aspect of the different centrality indices in the different layers of the Tweet network: it seems that there is no obvious correlation between the number of direct followers and their centrality with respect to various aspects of communication on Twitter, as detailed in the following. In the Mention layer, nodes 2, 59, and 96 are among the top 10 nodes, but their number of followers varies between as little as 322 (node 96) and 33,664 (!) friends (node 59). This is a very interesting result as the number of direct friends should be assumed to correlate strongly with the number of mentions or replies, but it is not necessarily the case, as can be seen here. Nodes 15 and 28 have a similar situation as the last cases. Although node 15 has a very large number of friends/followers (11,880) –about 40 times larger than the other– they stay among almost similar range of ranking positions with respect to mentioning the other users in their re-tweeted tweets. Similarly, but less extreme results can be seen on the Reply layer, where nodes 103, 26, and 46 show similar rankings despite the fact that node 103 has about 3 and 5 times more friends than nodes 26 and 46, respectively. Vice versa, nodes 40 and 46, both with about 500 friends/followers, show distinct behaviors, especially with respect to operators with a high *andness*. In Figure 4b, nodes 2 and 103 have

similar patterns of ranking and thus located in one category and similarly, nodes 28 and 15 placed in the top left group close to each other.

4 Summary

In this paper, we investigate the influence of the nodes in three different multiplex network data sets each of which contained a three-layer network and in each layer, multiple network processes of interest can occur. Since the centrality indices corresponding with these network processes result in conflicting rankings, we propose to use a fuzzy operator that scales between emphasizing the result of either *at least one* or *all* centrality indices. By comparing the curves for different values of β of one node in all layers of interest, the overall importance of a node for different network processes in different but related network structures can be explored. Then, using two proposed measures in a visualization, the overall ranking pattern of nodes can be analysed. For the air transportation network, we basically see two different behaviors: either, the airport has almost the same centrality for all network processes or it is a very influential node in one or two airlines and unimportant for the remaining one(s). In the second network data set, the centrality indices were much more conflicting that resulted in more different ranking behaviors. In the third network dataset we find that the number of direct followers is not necessarily correlated with other aspects of communication on Twitter and the exploration shows interesting individuals who are influential with respect to various, possible network processes despite their low number of direct followers. In general, the method reveals that centrality indices are not easily interchangeable because they produce quite different rankings. By correlating the new insights with external variables, it might even be possible to find out whether it is a better strategy to copy other peoples' behavior or to complement it, i.e. whether the important positions in a network are rather shared by more or less the same nodes or whether they are partitioned onto different nodes. The answer to this question will be left to future works.

References

- [1] Abufouda, M., Zweig, K.A.: Interactions around social networks matter: Predicting the social network from associated interaction networks. In: Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 142–145. IEEE/ACM (2014)
- [2] Battiston, F., Nicosia, V., Latora, V.: Structural measures for multiplex networks. *Physical Review E* **89**(3), 032,804 (2014)
- [3] Borgatti, S.: Centrality and network flow. *Social Networks* **27**(1), 55 – 71 (2005)
- [4] Cardillo, A., Gómez-Gardenes, J., Zanin, M., Romance, M., Papo, D., del Pozo, F., Boccaletti, S.: Emergence of network features from multiplexity. *Scientific reports* **3** (2013)
- [5] De Domenico, M., Lima, A., Mougél, P., Musolesi, M.: The anatomy of a scientific rumor. *Scientific Reports* **3**, 2980 (2013)

- [6] De Domenico, M., Solé-Ribalta, A., Omodei, E., Gómez, S., Arenas, A.: Ranking in inter-connected multilayer networks reveals versatile nodes. *Nature communications* **6**, 6868 (2015)
- [7] Filev, D., Yager, R.R.: Analytic properties of maximum entropy OWA operators. *Information Sciences* **85**(1), 11–27 (1995)
- [8] Freeman, L.: Centrality in social network, conceptual clarification. *Social Networks* **1**, 215–239 (1979)
- [9] Guimera, R., Mossa, S., Turtschi, A., Amaral, L.A.: The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. *Proceedings of the National Academy of Sciences* **102**(22), 7794–7799 (2005)
- [10] Keeling, M.J., Rohani, P.: *Modeling infectious diseases in humans and animals*. Princeton University Press (2008)
- [11] Kitsak, M., Gallos, L.K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H.E., Makse, H.A.: Identification of influential spreaders in complex networks. *Nature physics* **6**(11), 888–893 (2010)
- [12] Kivela, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. *Journal of Complex Networks* **2**(3), 203–271 (2014)
- [13] Koschützki, D., Lehmann, K.A., Peeters, L., Richter, S., Tenfelde-Podehl, D., Zlotowski, O.: *Network Analysis - Methodological Foundations*, chap. Centrality Indices, pp. 16–60. Springer Verlag (2005)
- [14] Koschützki, D., Lehmann, K.A., Tenfelde-Podehl, D., Zlotowski, O.: *Network Analysis - Methodological Foundations*, chap. Advanced Centrality Concepts, pp. 83–110. Springer Verlag (2005)
- [15] Lazega, E.: *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership*. Oxford University Press on Demand (2001)
- [16] Solé-Ribalta, A., De Domenico, M., Gómez, S., Arenas, A.: Centrality rankings in multiplex networks. In: *Proceedings of the 2014 ACM Conference on Web Science, WebSci '14*, pp. 149–155. ACM, New York, NY, USA (2014)
- [17] Tavassoli, S., Zweig, K.A.: Analyzing the activity of a person in a chat by combining network analysis and fuzzy logic. In: *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1565–1568. IEEE/ACM (2015)
- [18] Tavassoli, S., Zweig, K.A.: Most central or least central? how much modeling decisions influence a node's centrality ranking in multiplex networks. *arXiv preprint arXiv:1606.05468* (2016)
- [19] Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decision-making. *IEEE Transactions on systems, Man, and Cybernetics* **18**(1), 183–190 (1988)
- [20] Yager, R.R.: Quantifier guided aggregation using OWA operators. *International Journal of Intelligent Systems* **11**(1), 49–73 (1996)
- [21] Zadeh, L.A.: Fuzzy sets. *Information and Control* **8**(3), 338–353 (1965)

Preserving Sparsity in Dynamic Network Computations

Francesca Arrigo and Desmond J. Higham

Abstract Time sliced networks describing human-human digital interactions are typically large and sparse. This is the case, for example, with pairwise connectivity describing social media, voice call or physical proximity, when measured over seconds, minutes or hours. However, if we wish to quantify and compare the overall time-dependent centrality of the network nodes, then we should account for the global flow of information through time. Because the time-dependent edge structure typically allows information to diffuse widely around the network, a natural summary of sparse but dynamic pairwise interactions will generally take the form of a large dense matrix. For this reason, computing nodal centralities for a time-dependent network can be extremely expensive in terms of both computation and storage; much more so than for a single, static network. In this work, we focus on the case of dynamic communicability, which leads to broadcast and receive centrality measures. We derive a new algorithm for computing time-dependent centrality that works with a sparsified version of the dynamic communicability matrix. In this way, the computation and storage requirements are reduced to those of a sparse, static network at each time point. The new algorithm is justified from first principles and then tested on a large scale data set. We find that even with very stringent sparsity requirements (retaining no more than ten times the number of nonzeros in the individual time slices), the algorithm accurately reproduces the list of highly central nodes given by the underlying full system. This allows us to capture centrality over time with a minimal level of storage and with a cost that scales only linearly with the number of time points.

Francesca Arrigo (e-mail: francesca.arrigo@strath.ac.uk)✉ · Desmond J. Higham (e-mail: d.j.higham@strath.ac.uk)✉
University of Strathclyde, 16 Richmond St, Glasgow G1 1XQ,

The work of the authors was supported by the Engineering and Physical Sciences Research Council under grant EP/M00158X/1.

1 Introduction

In network science, centrality measures assign to each node a value that summarises some aspect of its relative importance. Such measures arose in the social sciences, but have now become very widely used by researchers who wish to summarise important features of large, complex networks [5, 14, 19]. Because matrix representations of networks are typically sparse, and because centrality measures usually involve the solution of linear systems or eigenvalue problems, it is feasible to compute centrality measures on a current desktop computer for networks with, say, a number of nodes in the millions.

Our focus in this work is the case of time-dependent network sequences [8]. Such data sets may be regarded as three-dimensional tensors, where, along with the (i, j) coordinates that capture pairwise connectivity, we also have a third coordinate that represents time [1]. These types of connections arise, for example, when we record human-human digital interaction through social media, telecommunication or physical proximity. In [7] the concept of a *dynamic communicability matrix* was introduced, which converted the time sequence of networks into a single two-dimensional array, with (i, j) element summarising the ability of node i to communicate with node j , using the time-dependent sequence of edges recorded in the data. From this matrix, it is straightforward to compute centrality measures:

- *dynamic broadcast centrality* takes large values for nodes that are effective at distributing information,
- *dynamic receive centrality* takes large values for nodes that are effective at gathering information.

In a case study on Twitter data, this approach was seen to be successful, in the sense of correlating well with the independent views of social media experts [10]. It was also found to outperform the crude alternative of simply aggregating all edges into a single static network that forgets the time-ordering of the interactions; see [12] for further discussion. Tests in [4, 13] also showed that dynamic broadcast centrality can be effective at quantifying the potential for the spread of disease across time-ordered interactions.

However, as we explain in the next section, the computation of dynamic broadcast centrality can be expensive in terms of both storage and computation, as a result of inevitable matrix fill-in as temporal information accumulates. Our overall aim here is to address this issue by deriving a new algorithm that delivers good approximations to the original dynamic broadcast centrality measure while retaining the benefits of the sparsity present in the time slices.

We note that other approaches to computation of node centrality for time-dependent networks have been put forward. For example, [15, 16, 17] made use of paths rather than walks, which, for our purposes, leads to an infeasibly expensive algorithm. In [18] a block-matrix approach was suggested which allows centrality measures for static networks to be applied. However, as mentioned in [12], that formulation does not fully respect the arrow of time.

2 Background and Notation

In this section we recall some definitions and notation that will be used throughout. Let $t_0 < t_1 < \dots < t_M$ be an ordered sequence of time points and let $\{\mathcal{G}^{[k]}\}_{k=0}^M = \{(\mathcal{V}^{[k]}, \mathcal{E}^{[k]})\}$ be a time-ordered sequence of unweighted graphs defined over n nodes. A graph is said to be unweighted when all its edges have the same weight, which can thus be assumed to be unitary. Consider the adjacency matrices $\{A^{[k]}\}_{k=0}^M = \{(a_{ij}^{[k]})\} \in \mathbb{R}^{n \times n}$ associated with these graphs at times $\{t_k\}_{k=0}^M$, whose entries are defined as

$$a_{ij}^{[k]} = \begin{cases} 1 & \text{if } (i, j) \in \mathcal{E}^{[k]} \\ 0 & \text{otherwise.} \end{cases}$$

In [7] the concept of a *dynamic walk of length p* was introduced to extend to the temporal case the well-known concept of a walk of length p in static networks. Loosely, we have a (possibly repeated) sequence of $p + 1$ nodes connected by edges that appear in a suitable order. More precisely, a dynamic walk of length p from node i_1 to node i_{p+1} consists of a sequence of nodes i_1, i_2, \dots, i_{p+1} and a sequence of times $t_{r_1} \leq t_{r_2} \leq \dots \leq t_{r_p}$ such that $a_{i_m i_{m+1}}^{[r_m]} \neq 0$ for $m = 1, 2, \dots, p$. We stress that more than one edge can share a time slot, and that time slots must be ordered but do not need to be consecutive.

The concept of dynamic walk was used to motivate the definition of the *dynamic communicability matrix*

$$Q^{[M]} = (I - \alpha A^{[0]})^{-1} (I - \alpha A^{[2]})^{-1} \dots (I - \alpha A^{[M]})^{-1}, \quad (1a)$$

which can be defined equivalently via the iteration

$$Q^{[k]} = Q^{[k-1]} (I - \alpha A^{[k]})^{-1}, \quad k = 0, 1, \dots, M, \quad (1b)$$

where $Q^{[-1]} = I$ is the identity matrix of order n , $0 < \alpha < 1/\rho^*$, and $\rho^* = \max_{k=0:M} \{\rho(A^{[k]})\}$ is the largest spectral radius among the spectral radii of the matrices $\{A^{[k]}\}$. Here the free parameter α plays the same role as in the classical Katz centrality measure for static networks [5, 9, 14]. For simplicity, our notation does not explicitly record the dependence of Q upon α .

To avoid overflow in the computations, a normalisation step $Q \mapsto Q/|Q|$ should follow each iteration in (1b). Throughout this work we use the Euclidean norm.

The requirement $\alpha < 1/\rho^*$ ensures that the resolvents in (1a) exist and can be expanded as $(I - \alpha A^{[k]})^{-1} = \sum_{p=0}^{\infty} (\alpha A^{[k]})^p$. It follows that the entries of $Q^{[k]}$ provide a weighted count of the dynamic walks between any two nodes in the networks using the ordered sequence of matrices $A^{[0]}, A^{[1]}, \dots, A^{[k]}$, weighting walks of length p by a factor α^p . Hence, $(Q^{[k]})_{ij}$ is an overall measure of the ability of node i to send messages to node j .

Using the dynamic communicability matrix one can define and compare the broadcast and receive centrality of nodes by taking row and column sums of the matrix $Q^{[M]}$, respectively. The *broadcast centrality* of node i is defined as $b_i^{[M]} := \mathbf{e}_i^T Q^{[M]} \mathbf{1}$, where $\mathbf{e}_i \in \mathbb{R}^n$ is the i th column of I , the superscript “ T ” denotes transposition, and $\mathbf{1} \in \mathbb{R}^n$ is the vector of all ones. Similarly, the *receive centrality* of node j is defined as $r_j^{[M]} := \mathbf{1}^T Q^{[M]} \mathbf{e}_j$. It is straightforward to show that the latter satisfies a

lower-dimensional, vector-valued iteration given by

$$\mathbf{r}^{[k]} := \mathbf{1}^T \mathbf{Q}^{[k]} = \mathbf{r}^{[k-1]}(I - \alpha A^{[k]})^{-1}, \quad k = 0, 1, \dots, M,$$

with $\mathbf{r}^{[-1]} = \mathbf{1}$. The receive centrality of the nodes can thus be updated at each step by solving a single sparse linear system whose coefficient matrix is the latest network time slice. In particular, this means that we do not need to store and update the full matrix $\mathbf{Q}^{[k]}$ to recover the receive centrality of nodes at level k . By contrast, to compute the broadcast centrality vector, $\mathbf{b}^{[M]} = \mathbf{Q}^{[M]}\mathbf{1}$, we need access to the current dynamic communicability matrix at each step. Intuitively, this difference arises because,

- given a summary of how much information is flowing *into* each node, we can propagate this information forward when new edges emerge: receive centrality cares about where the information *terminates*, but
- a summary of how much information is flowing *out of* each node cannot be straightforwardly updated when new edges emerge: broadcast centrality cares about where the information *originates*.

Our focus here is on the natural setting where data is processed sequentially, with the centrality scores being updated as each new time slice $A^{[k]}$ arrives. As confirmed in Section 4 on a real data set, we then face a fundamental issue with the use of the dynamic communicability matrix: although the time slices are typically sparse, $\mathbf{Q}^{[k]}$ generally evolves into a dense matrix. At this stage, computing dynamic communicability from (1b) requires us to store a full $O(n^2)$ matrix and solve at each subsequent time point a corresponding full linear system. In the next section, we therefore develop and justify an approximation where matrix fill-in is controlled so that the benefits of sparse matrix storage and computation are recovered.

3 Sparsification

To create a sparse approximation, $\widehat{\mathbf{Q}}^{[k]}$, to the dynamic communicability matrix, $\mathbf{Q}^{[k]}$, we first observe that the original iteration (1b) includes some traversals that are not very meaningful, e.g., repeated cycles $i \rightarrow j \rightarrow i \rightarrow j \rightarrow i \rightarrow j$ using the same undirected edge at the same time point. We thus use an “at most one edge per time point” alternative to (1b) so as to avoid considering these types of walks and similar ones:

$$\widehat{\mathbf{Q}}^{[k]} = \widehat{\mathbf{Q}}^{[k-1]}(I + \alpha A^{[k]}), \quad k = 0, 1, \dots, M, \quad (2)$$

with $\widehat{\mathbf{Q}}^{[-1]} = I$. As discussed in [7], this matrix product can be interpreted in terms of network combinatorics; at each time step a dynamic traversal can either wait, as described by the identity matrix I , or take a current edge, as described by latest adjacency matrix, $A^{[k]}$. In the latter case, the length of the walk (i.e., the number of edges used) has increased by one, and thus we multiply the corresponding matrix by α . An alternative interpretation is that we are using a second order Taylor approximation for each of the resolvents appearing in (1b). This simplification is likely to be reasonable when either (a) α is chosen to be small, so that short walks are favoured, or (b) the

powers of $A^{[k]}$ do not grow rapidly with k (which is typically the case for sparse matrices).

As the time index k increases in (2) the number of nonzeros cannot decrease, and the matrix $\widehat{Q}^{[k]}$ will generally fill in. In order to produce a sparse approximation we will proceed iteratively. At each step we threshold the matrix at a level θ_k —this type of approach has been widely used in large scale machine learning, data mining, and signal processing; see, e.g., [2, 3] and references therein. Hence, for $k = 0, 1, \dots, M$ we redefine the iteration to be

$$\widehat{Q}^{[k]} = \frac{\lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k}}{\| \lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k} \|_2}, \quad (3)$$

where $\widehat{Q}^{[-1]} = I$ and for any nonnegative matrix $C = (c_{ij})$, the matrix $\lfloor C \rfloor_{\theta_k}$ arises from setting to zero all entries where $c_{ij} \leq \theta_k$.

Remark 3.1. The matrices $\{\widehat{Q}^{[k]}\}_{k=0}^M$ are non-negative by construction.

3.1 A little twist

From a network science perspective, the approach just presented has a strong limitation. Imagine a user i of Twitter who remains inactive for a long time after each tweet. After such inactivity, the thresholding may zero out all entries in the i th row of one of the matrices $\widehat{Q}^{[k]}$. From that time, the i th row of the matrices appearing in (3) will always be zero, and no subsequent activity of node i will be registered by this approach.

To mitigate pathological behaviour of this type, we modify (3) so as to keep track at each step of the behaviour of those nodes corresponding to zero rows in the iteration matrix. Our final version of the iteration goes as follows:

$$\widehat{Q}^{[k]} = \lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k} + m_k \mathcal{A}^{[k]}, \quad k = 0, 1, \dots, M, \quad (4)$$

followed by normalisation, where $\widehat{Q}^{[-1]} = I$, m_k is the smallest nonzero entry of $\lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k}$, $\mathcal{A}^{[k]} = \alpha W^{[k]} A^{[k]}$, and $W^{[k]} = \text{diag}(w_1, w_2, \dots, w_n) \in \mathbb{R}^{n \times n}$ is a diagonal matrix whose entries are

$$w_i = \begin{cases} 1 & \text{if } \mathbf{e}_i^T \lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k} \mathbf{1} = 0 \\ 0 & \text{otherwise.} \end{cases}$$

The matrix $\mathcal{A}^{[k]}$ keeps track of those edges that appear at step k and would otherwise get lost. Indeed, the matrix product $W^{[k]} A^{[k]}$ returns a matrix that has nonzero entries (if any) only in the rows corresponding to those nodes that have either been inactive until step k or have broadcast very little information (which thus was thresholded in a previous iteration). The penalisation by α is added because we are taking one hop in the network. Finally, the multiplication by m_k comes from the fact that a poor choice of the parameter α may compromise the results. Indeed, the entries of $\mathcal{A}^{[k]}$ may be too large with respect to those appearing in $\lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k}$, thus leading to a complete reshaping of the rankings. We refer the reader to Section 4 for an example of this issue.

Remark 3.2. It is possible for the contribution added by $m_k \mathcal{A}^{[k]}$ to be zero. This happens when the zero rows in $\lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k}$ correspond to nodes that are not broadcasting information at step k .

Remark 3.3. Note that if $A^{[k]} = 0$ for some k , then $\widehat{Q}^{[k]} = \widehat{Q}^{[k-1]}$, just as $Q^{[k]} = Q^{[k-1]}$.

3.2 On the thresholding parameters

The thresholding parameters $\{\theta_k\}$ are a key part of the sparsification process. Before explaining how we select these values in applications, we first describe the types of contributions that are removed from the approximation to the dynamic communicability matrix when the thresholding is performed. There are two key circumstances where the thresholding has an effect:

- the value of α^p dominates the contribution given by the products of the adjacency matrices, i.e., there are not too many walks of length p between the two nodes under consideration;
- the information has not moved from a certain node for a long time and the normalisation step has made the corresponding contribution smaller than the other entries.

In both cases, we are dismissing information that has little potential, as it is not diffused much. Clearly, an over-stringent selection of the parameters θ_k may lead to an excessive penalisation of these two types of behaviours. Our strategy is to make an initial choice for the maximum number of nonzeros that we will allow in the matrices $\widehat{Q}^{[k]}$, for $k = 0, 1, \dots, M$. Then, as the iteration proceeds, the thresholding value θ_k is chosen so as to make $\lfloor \widehat{Q}^{[k-1]}(I + \alpha A^{[k]}) \rfloor_{\theta_k}$ have approximately this desired level of sparsity.

We point out that the maximum number of nonzeros one wants to allow has to be at least $n + \text{nnz}(A^{[0]})$, where $\text{nnz}(A^{[0]})$ is the number of nonzeros in the matrix $A^{[0]}$. Consequently, $\theta_0 < \alpha$. Indeed, if this is not the case, then we will have $\theta_k \geq \alpha$ for all k and therefore that $\widehat{Q}^{[k]} = I$ for all k .

3.3 Cost Comparison

We are now in a position to quantify, at least approximately, the computational benefits of using $\widehat{Q}^{[k]}$ in (4) rather than the exact matrix $Q^{[k]}$ in (1b) to compute dynamic broadcast communicability. Because the exact representation $Q^{[k]}$ becomes full in general, it follows that:

- We have reduced storage requirements by a factor of n .
- We have reduced the dominant computational task at each time step from solving n sparse linear systems to multiplying two sparse matrices. For general complex networks with no exploitable structure, if a standard iterative scheme is used to solve a sparse linear system, each matrix vector multiplication will cost $O(n)$

and thus the total cost to compute $Q^{[k]}$ by solving n such linear systems will be at least $O(n^2)$. Instead, the overall cost of computing the product of $\widehat{Q}^{[k-1]}$ times $A^{[k]}$ is $O(n)$, if we assume that there is a fixed number of active nodes at each time point. Thus, the cost has been reduced by a factor of n .

3.4 Comparing top K lists

The main goal of this work is to match the broadcast ranking of the nodes in an evolving network using a sparse approximation to the dynamic communicability matrix. As usual in network science, we are not interested in matching exactly the rankings of all nodes in the network, but rather to accurately capture the top $K \ll n$ most influential broadcasters. Although there is no perfect way to summarise and compare rankings, it is clear that generic correlation coefficients like Pearson's correlation coefficient or Kendall's tau have the major drawback in this context that they treat entire vectors, and hence all network nodes.

In order to compare the top K entries of two ranking vectors, an appropriate index is the *intersection similarity* [6]. This quantity is defined as follows: given two ranked lists x and y , consider the top K entries of each, which we denote x_K and y_K , respectively. Then, the top K intersection similarity between x and y is defined as

$$\text{isim}_K(x, y) = \frac{1}{K} \sum_{i=1}^K \frac{|x_i \Delta y_i|}{2i}, \quad (5)$$

where Δ is the symmetric difference operator between two sets and $|S|$ denotes the cardinality of the set S . When the sequences contained in x and y are completely different, the intersection similarity between the two is maximum and equals 1. On the other hand, when $\text{isim}_K(x, y) = 0$ for all K , then the two lists are identical.

It happens sometimes that the two lists differ in the *order*, but not in the *set of labels* of the nodes appearing in them. Behaviour of this type can be easily spotted by looking at the quantity

$$\ell_K(x, y) = \frac{|x_K \Delta y_K|}{2K}, \quad K = 2, 3, \dots$$

If $\ell_K(x, y) = 0$ for some K we know that x_K and y_K are permutations of the same set of nodes.

4 Numerical tests

We have tested the new algorithm on large scale data sets involving email, voice call and on-line social interaction, and with various values of the parameter α . Due to space limitations we give representative results with the email data set Enron [11]. Here, a directed edge from node i to node j indicates that at least one message was sent from i to j in a one day period, including `to`, `cc`, and `bcc`. We have information over 1138 days starting 11 May 1999 for 151 Enron employees, Many of the adjacency matrices are empty, meaning that there are days during which no

emails are sent. The largest spectral radius is $\rho^* = 4.17$, thus the upper limit for α is 0.24.

We allowed for a number of nonzeros proportional to $N = c\bar{n}$, where $\bar{n} = n + \frac{1}{M+1} \sum_{k=0}^M \text{nnz}(A^{[k]})$ and $c = 10$. This is motivated by our aim to work only with matrices whose sparsity level is compatible with that of the individual network time slices. Further testing has shown that the performance is not sensitive to c .

4.1 Adaptive Scaling

Before testing the performance of (4), in this subsection we discuss the effect of including the multiplication by m_k . In Section 3 we argue that setting $m_k \equiv 1$ for all $k = 0, 1, \dots, M$ in (4) may lead to poor results. Clearly, this is not always the case, but, as we will see here, this choice together with a compounding choice of the downweighting parameter α , may result in a complete misplacement of the top ranked broadcasters in the network.

We compute the broadcast centrality vector $Q^{[M]}\mathbf{1}$ and our approximation vector $\widehat{Q}^{[M]}\mathbf{1}$ for seven different values of the downweighting parameter:

$$\alpha = \frac{0.01}{\rho^*}, \frac{0.1}{\rho^*}, \frac{0.25}{\rho^*}, \frac{0.5}{\rho^*}, \frac{0.75}{\rho^*}, \frac{0.85}{\rho^*}, \frac{0.9}{\rho^*}.$$

Figure 1 displays the evolution of the intersection similarity between the top $K = 1, 2, \dots, 20$ entries of the vectors $Q^{[M]}\mathbf{1}$ and $\widehat{Q}^{[M]}\mathbf{1}$ versus K for the different values of α . The left plot contains the results when $m_k \equiv 1$, while the right plot contains the results when m_k is adapted by setting it to be equal to the smallest nonzero entry of the matrix $[\widehat{Q}^{[k-1]}(I + \alpha A^{[k]})]_{\theta_k}$ at each iteration.

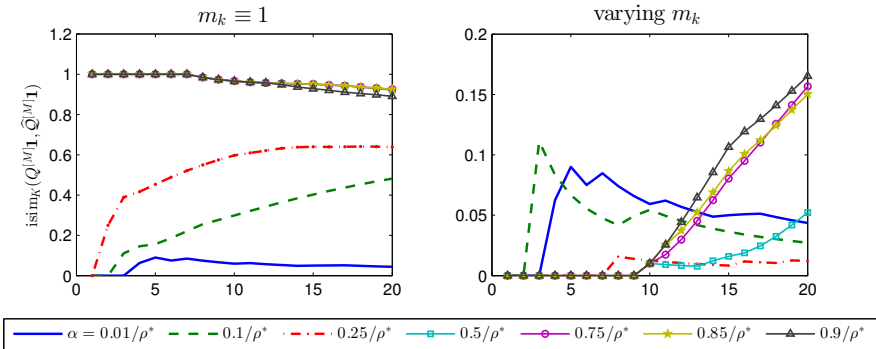


Fig. 1: Evolution of the intersection similarity $\text{isim}_K(Q^{[M]}\mathbf{1}, \widehat{Q}^{[M]}\mathbf{1})$ versus K , for different choices of the downweighting parameter α . Left: $m_k \equiv 1$. Right: m_k is set at each iteration as the smallest nonzero entry of $[\widehat{Q}^{[k-1]}(I + \alpha A^{[k]})]_{\theta_k}$. Note the difference in vertical axis range.

These results show that when $m_k \equiv 1$ the intersection similarity between the two vectors can be maximum even when comparing only a few top ranked nodes for α as

Table 1: Top 10 ranked nodes: exact, approximate and with aggregate out-degree.

$Q^{[M]}\mathbf{1}$	48 67 147 73 13 50 137 49 9 139
$\widehat{Q}^{[M]}\mathbf{1}$	48 67 147 73 13 50 137 49 9 139
out-degree	67 50 141 13 48 69 107 147 73 70

small as $0.5/\rho^*$. The right hand plot in the figure shows how an adaptive choice of m_k can work successfully over a wide range of α choices.

4.2 Centrality Approximation

We now assess the effectiveness of iteration (4) at approximating the broadcast centrality rankings. Using $\alpha = 0.01$, the number of nonzero entries in the dynamic communicability matrix is $\text{nnz}(Q^{[M]}) = 21097$. Note that $n^2 = 22801$, so the matrix is 92.5% full. Figure 2 scatter plots the resulting approximation to the broadcast and receive centrality vectors against $Q^{[M]}\mathbf{1}$ and $\mathbf{1}^T Q^{[M]}$, respectively. We observe a good linear correlation at the high end for both cases, indicating that our method correctly identifies important nodes. The number of nonzeros in the final approximation matrix $\widehat{Q}^{[M]}$ is = 1676, so the level of sparsity has been reduced to around 7.4%.

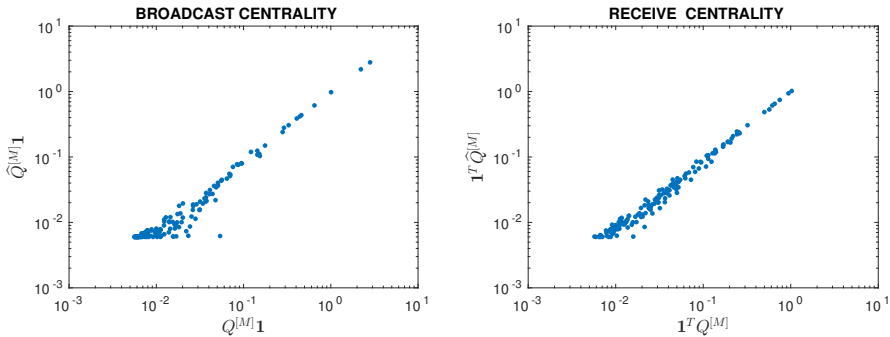


Fig. 2: Comparison of exact (horizontal) and approximate (vertical) centralities.

In Table 1 we list the top 10 ranked nodes according to the broadcast centrality. The first row contains the true result, obtained by ranking the nodes according to $Q^{[M]}\mathbf{1}$; in the second row we list the top 10 broadcasters according to the ranking derived from $\widehat{Q}^{[M]}\mathbf{1}$ and, finally, the last row displays the result obtained when the nodes are ranked according to their aggregate out-degree: $\sum_{k=0}^M A^{[k]}\mathbf{1}$. As $\alpha \rightarrow 0$, the ranking obtained using the dynamic communicability matrix approaches that obtained using the aggregate out-degree; see, e.g., [4, 7]. Clearly, however, $\alpha = 0.01$ is not close enough to zero for this effect to be observed.

Tables 2-3 contain the values of $\text{isim}_K(Q^{[M]}\mathbf{1}, \widehat{Q}^{[M]}\mathbf{1})$ for $K = 1, 2, \dots, 20$ and $\ell_K(Q^{[M]}\mathbf{1}, \widehat{Q}^{[M]}\mathbf{1})$ for $K = 2, 3, \dots, 20$. We see that the new method correctly orders the top 11 broadcasters in the network and correctly identifies the top 20.

Table 2: Intersection similarity between the top $K = 1, 2, \dots, 20$ ranked nodes in $Q^{[M]}\mathbf{1}$ and $\widehat{Q}^{[M]}\mathbf{1}$.

K	1	2	3	4	5	6	7	8	9	10
isim_K	0	0	0	0	0	0	0	0	0	0
K	11	12	13	14	15	16	17	18	19	20
isim_K	0	0.01	0.02	0.03	0.03	0.03	0.03	0.03	0.03	0.03

Table 3: Evolution of $\ell_K(Q^{[M]}\mathbf{1}, \widehat{Q}^{[M]}\mathbf{1})$ for $K = 2, 3, \dots, 20$.

K	2	3	4	5	6	7	8	9	10	
ℓ_K	0	0	0	0	0	0	0	0	0	
K	11	12	13	14	15	16	17	18	19	20
ℓ_K	0	0.08	0.15	0.14	0.07	0	0.06	0	0.05	0

5 Conclusions

Time-dependency adds an extra dimension to network science computations, potentially causing a dramatic increase in both storage requirements and computation time. In the case of Katz-style centrality measures, which are based on the solution of linear algebraic systems, allowing for the arrow of time leads naturally to full matrices that keep track of all possible routes for the flow of information. Such a build-up of intermediate data can make large-scale computations unfeasible. In this work, we derived a sparsification technique that delivers accurate approximations to the full-matrix centrality rankings, while retaining the level of sparsity present in the network time-slices. With the new algorithm, as we move forward in time the storage cost remains fixed and the computational cost scales linearly, so the overall task is equivalent to solving a single Katz-style problem at each new time point.

References

- [1] Acar, E., Dunlavy, D.M., Kolda, T.G.: Link prediction on evolving data using matrix and tensor factorizations. In: ICDMW'09: Proceedings of the 2009 IEEE International Conference

- on Data Mining Workshops, pp. 262–269 (2009). DOI 10.1109/ICDMW.2009.54
- [2] Achlioptas, D., Karnin, Z.S., Liberty, E.: Near-optimal entrywise sampling for data matrices. In: C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K.Q. Weinberger (eds.) *Advances in Neural Information Processing Systems 26*, pp. 1565–1573. Curran Associates, Inc. (2013). URL <http://papers.nips.cc/paper/5036-near-optimal-entrywise-sampling-for-data-matrices.pdf>
 - [3] Arora, S., Hazan, E., Kale, S.: A fast random sampling algorithm for sparsifying matrices. In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 272–279. Springer (2006)
 - [4] Chen, I., Benzi, M., Chang, H.H., Hertzberg, V.S.: Dynamic communicability and epidemic spread: a case study on an empirical dynamic contact network. *Journal of Complex Networks* (2016). DOI 10.1093/comnet/cnw017. URL <http://comnet.oxfordjournals.org/content/early/2016/06/07/comnet.cnw017.abstract>
 - [5] Estrada, E.: *The Structure of Complex Networks*. Oxford University Press, Oxford (2011)
 - [6] Fagin, R., Kumar, R., Sivakumar, D.: Comparing top k lists. *SIAM Journal on Discrete Mathematics* **17**(1), 134–160 (2003)
 - [7] Grindrod, P., Parsons, M.C., Higham, D.J., Estrada, E.: Communicability across evolving networks. *Physical Review E* **83**(4), 046,120 (2011)
 - [8] Holme, P., Saramäki, J.: Temporal networks. *Physics Reports* **519**, 97–125 (2011)
 - [9] Katz, L.: A new status index derived from sociometric analysis. *Psychometrika* **18**(1), 39–43 (1953)
 - [10] Laffin, P., Mantzaris, A.V., Grindrod, P., Ainley, F., Otley, A., Higham, D.J.: Discovering and validating influence in a dynamic online social network. *Social Network Analysis and Mining* **3**, 1311–1323 (2013)
 - [11] Leskovec, J.: SNAP: Network dataset. <https://snap.stanford.edu/data/>
 - [12] Mantzaris, A.V., Higham, D.J.: Asymmetry through time dependency. *Eur. Phys. J. B* **89**(3), 71 (2016). DOI 10.1140/epjb/e2016-60639-0. URL <http://dx.doi.org/10.1140/epjb/e2016-60639-0>
 - [13] Mantzaris, A.V., Higham, D.J.: Dynamic communicability predicts infectiousness. In: P. Holme, J. Saramäki (eds.) *Temporal Networks*, pp. 283–294. Springer, Berlin (2103)
 - [14] Newman, M.E.J.: *Networks: An Introduction*. Oxford University Press, Oxford (2010)
 - [15] Tang, J., Musolesi, M., Mascolo, C., Latora, V.: Temporal distance metrics for social network analysis. In: *Proceedings of the 2nd ACM SIGCOMM Workshop on Online Social Networks (WOSN09)*. Barcelona (2009)
 - [16] Tang, J., Musolesi, M., Mascolo, C., Latora, V.: Characterising temporal distance and reachability in mobile and online social networks. *SIGCOMM Comput. Commun. Rev.* **40**, 118–124 (2010)
 - [17] Tang, J., Scellato, S., Musolesi, M., Mascolo, C., Latora, V.: Small-world behavior in time-varying graphs. *Physical Review E* **81**, 05,510 (2010)
 - [18] Taylor, D., Myers, S.A., Clauset, A., Porter, M.A., Mucha, P.J.: Eigenvector-based centrality measures for temporal networks (2015). ArXiv:1507.01266
 - [19] Wasserman, S., Faust, K.: *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge (1994)

Flows of Knowledge in Citation Networks

Benjamin Renoust, Vivek Claver and Jean-François Baffier

Abstract Knowledge is created and transmitted through generation. Innovation is often seen as a generative process from collective intelligence, but how does innovation emerges from the blending of accumulated knowledge, and from which path an innovation mostly inherit? A citation network can be seen as a perfect example of a generative process leading to innovation. Inspired by the notion of “stream of knowledge”, we propose to look at the question of production of knowledge under the lens of DAGs. Although many works look for the evaluation of publications, we propose to look for production of knowledge within a framework for analyzing DAGs. In this framework inspired by the work of Strahler, we can also account for other well known measures of influence such as the h -index. We propose then to analyze flows of influence in a citation networks as an ascending flow. We propose an efficient dynamic algorithm for integration with modern graph databases, conducting our experiment with the Arxiv HEP-TH dataset. Our results validate the use of DAG flows for citation flows and show evidence of the relevance of the h -index.

1 Introduction

From the ancient times, knowledge passes from individuals to others leading at each step to more discoveries and innovations. In modern times, with the industrialization of research, it has become key to track this production of knowledge [16, 27]. Indeed, it is important for the newly produced innovation to state on which ground it stands, so peers can judge of the quality of the proposed innovation. An innovation must cite

Benjamin Renoust (e-mail: renoust@nii.ac.jp) · Vivek Claver
National Institute of Informatics & JFLI CNRS UMI 3527, Tokyo, Japan

Vivek Claver (e-mail: vivek.claver@berkeley.edu) · Jean-François Baffier
University of California Berkeley, Berkeley, USA

Jean-François Baffier (e-mail: jf_baffier@nii.ac.jp)
JST-ERATO Kawarabayashi Large Graph project, Tokyo, JAPAN,

its influential sources to give credit to the work it was inspired from and to state its differences with the competing methods. This is one principle at the heart of the peer reviewing system enabling and validating the publication of new knowledge.

This process of citing sources is very important because it makes explicit the transmission of knowledge from prior works to an innovation [5] — and we can consider each new scientific publication as a container of an innovation. Thankfully, this production of scientific knowledge can be easily captured in a citation graph. In this graph, nodes are publications citing other publications. This citation relationship is oriented and corresponds to a borrowing or derivation of knowledge, and we suspect that the impact of a publication can be captured in this graph. The production of knowledge would then be represented as a growing process in a dynamic network.

Key for countries and organizations in modern science, the study of the production of knowledge is mostly considered from partial indicators to establish rankings and compare scientists. This gave rise to the development of many measures deriving from sociometrics [28] including age, field, and other cues. Three major indicators are often used: the number of citations, the impact factor [25] (which is a time-related average number of citations of a collection) and the h -index [19]. These are popular indicators used for the evaluation of scientists, however they can be subject to controversy [24] and are designed to reflect only the productivity of a scientist rather than measuring the production of knowledge.

One reason these indicators' popularity is their simplicity in terms of computing. However, when previous network analysis was seen as too complex to deploy, modern graph databases have now grown to ease the analysis of dynamic networks [7]. Inspired by the seminal work from Strahler [26] and from Hirsh [19] we propose to bring a fresh look at the production of knowledge based on the analysis of flows in Directed Acyclic Graphs (DAGs). This view is not limited to the production of indicators but allows a more in-depth analysis of the process and diffusion of knowledge. The traditional indicators are very effective and it is important that our framework allows to establish them, while being easily extended.

We first introduce the Strahler numbers [26] and the h -index [19] in a generalized flow framework, and how those two notions belong to one greater notion of flow, and introduce our ascending flow – modeled on the notion of flow of knowledge. We will then discuss parameters of this ascending flow to put it in relation with classical measures. We propose a dynamic algorithm that allows for quick update. We finally run experiments on a publicly available dataset, the ArXiv HEP-TH [15].

2 Related works

The study of the production and transmission of knowledge has attracted quite many scholars in the domains of social and economical science [17], with for example a focus on the population at the origin of production [29], and of transmission to business [14]. These studies come *a posteriori* when observing controlled domains, with well known sociometric indicators. We are instead interested in the modeling of the production and diffusion of knowledge.

Many interesting attempts for modeling the production and diffusion of knowledge are actually focused on the producer of knowledge themselves, such as in multi-agent simulation [9, 10]. In these models, the agents are actually interacting to produce knowledge, and the properties of the resulting interaction network of agents are the focus of analysis. The agents can actually be tuned to produce different resulting networks, simulating real world policies [23]. Even on real social networks, the topology of the networks of the people producing knowledge is the main focus of complex network research [11], because the focus is often to maximize diffusion in such network [1]. In contrast, our focus is on the information produced itself and how it relates to previous works.

A good model for this is the citation graph. It mostly apply to academic research, but have found its way in complex network research. Numerous works actually focus on communities [8], and the characterization of the dynamics of the citation graphs [15]. The closest to the spirit of our research would be the work by Hummon and Dereian [21] who studied the main paths in the citation network in order to extract backbones and areas of interest. The question of the efficient implementation of these cues has been the focus of a previous contribution [4]. An extension of Hummon and Dereian's original work has actually been applied to the study of the development of the h -index [22]. These methods are focused on the path produced by citations and use them as a base for bibliometrics, without capturing the global flow of information. We propose in contrast a natural interpretation of flows in DAGs that can easily capture the same measures used for main path analysis.

One of the most cited work in scientometrics is the *Hirsch index* [19], globally known as the h -index. It originally applies to the authors, and is designed to measures both the quantity and the quality of the authors' production. It was rapidly followed by numerous variants and extensions [28]. The most famous possibly is the g -index of Egghe [13] that is the largest number such that the g articles with the most citations receive at least a total of g^2 , averaging the importance of each article. Hirsch [20] proposes a more restrictive version called \bar{h} -index, normalized to domain or age. Other variants could be mentioned (such as Bucur *et al.* [6]), but each is designed with specific goals. All-in-all, h -index based measures are measures to analyze the productivity of researchers, but do not allow for the in-depth analysis of production, in contrary to main path analysis approaches.

Our work roots its contribution in the analysis of flows in DAGs. Traditional max-flow approaches are quite far from what we define here, because nodes are always sources of information and edges have infinite capacities — we may be closer to multicommodity flows [2]. Instead, we mostly take our inspiration from a different notion of flows, in river streams, as defined by Strahler [26]. Limited to binary trees, this notion has seen a few extensions [3, 12, 18] with applications to graph visualization. These versions use flows to highlight and extract most relevant paths in DAGs and trees and relatively place elements one to another. We will use this approach and adapt it to the production of knowledge.

In this work we propose to join the different views on knowledge production in a recursive framework. In section 3, we place in this framework different measures such as the h -index and Strahler number. Section 4 introduces our proposition of

a flow that captures the production of knowledge: the ascending flow. Finally, we provide experimental comparisons on the ArXiv HEP-TH dataset in section 5.

3 Preliminaries

We consider in our setting a citation graph $G = (V, E)$ in which a node $v \in V$ represents a publication, and a directed edge, hereafter an *arc*, $e(a, b) \in E$ is created when the article a cites an article b . We consider the graph as being directed acyclic (or DAG), although real-world data may introduce cycles, this is a marginal case that we will discard in our study.

In this setting, an author, a journal, proceedings or books can be modeled as collections of publications. Hence, by observing the collective impact of the collection we can characterize the influence this set of publications. In other words, in our citation graph formalism, collections are only sink nodes that can be sourced from the publications themselves. In this work, we will focus on measuring the impact of individual publications only, that can be trivially reported to authors and collections.

Definition 3.1. For a publication c , its neighborhood $\mathcal{N}(c)$ is the set of all the publications referring to c . The size of $\mathcal{N}(c)$ is simply its in-degree $d^-(c)$.

From its definition, the h -index applies in general trees of depth 3 and can actually be seen as a modified version of the Extended Strahler numbers [3], which generalize Strahler numbers [26] — limited to binary trees — to general trees. In this modification, a root node (*e.g.* an author) does not increase from his maximum valued nodes, but instead gets weighted by the maximum Extended Strahler number of his direct descendants (*i.e.* the publications).

Strahler numbers have been designed to define the size of river streams based on a hierarchy of dependent streams. Transmission of knowledge is very similar in that sense with publications being tributary to prior works they inherit from, and becoming in turn sources for later works — the h -index then captures the latter quantity. However, we want a finer measure which could capture the impact of a publication across all citations it generated.

We defined above our citations graphs to be DAGs, and fortunately, Strahler numbers have also been extended to DAGs [12, 18]. Herman *et al.* [18] proposes a generic framework to compute the importance K of nodes in DAGs — including Strahler numbers — such as:

$$K(v) = K(\mathcal{N}(v)) = \begin{cases} c, & \text{if } \mathcal{N}(v) = \emptyset \\ F(K(s_1), \dots, K(s_p)) & s_i \in \mathcal{N}(v) \text{ o.w.} \end{cases} \quad (1)$$

c designates a constant for terminal cases (leafs, often $c = 1$), F is an application of the neighborhood of v . s_i represents the successors (or a_i ancestors) of node v . This framework is nothing but a generic recursive framework, but it allows us to redefine in it other measures. In this context, counting the number of citations would only require to modify the application $F(\mathcal{N}(v))$, such as $F(\mathcal{N}(v)) = |\mathcal{N}(v)| = d^-(v)$. Similarly, the Strahler number of a node v is then defined as:

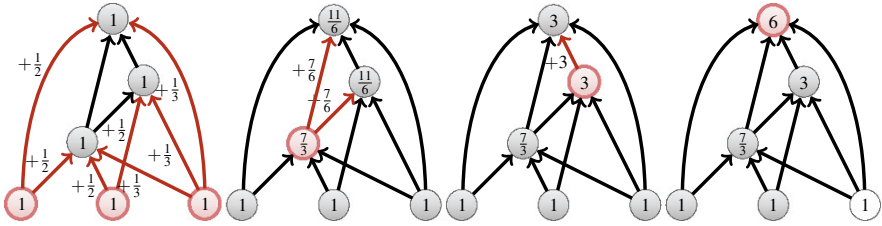


Fig. 1: Ascending flow algorithm: step by step

$$F(\mathcal{N}(v)) = \begin{cases} 1, \text{ if } d^-(v) = 0 \\ \max(K(s_1), \dots, K(s_p)) + \begin{cases} p - 1 \text{ if all values } K(s_i) \text{ are equal} \\ p - 2 \text{ otherwise} \end{cases} \end{cases} \quad (2)$$

The application for the h -index then becomes:

$$F(\mathcal{N}(v)) = \begin{cases} 0, \text{ if } d^+(v) = 0 \\ 1, \text{ if } d^+(v) = 1 \\ \max(K(k_1), \dots, K(s_p)) \mid |\{K(s_j)\}| = n, \text{ with } K(s_j) = n \end{cases} \quad (3)$$

Strahler numbers, number of citations, and h -index impose a discrete limit in depth which is conceptually an issue — there is no reason not to look for all the extended consequences of a publication. Instead, Herman *et al.* [18] propose in their framework a *Flow metric* for DAGs to emphasize the distribution of information to their successor such as:

$$F(\mathcal{N}(v)) = \begin{cases} 1, \text{ if } d^-(v) = 0 \\ \sum_i K(a_i) / d^-(a_i) \text{ o.w.} \end{cases} \quad (4)$$

In which a_i represents the ancestors of v (instead of the successors k_i). Note that this defines a *descending* flow measure which captures how much information all nodes in the network receive from a root node v , but does not give credit to v for its production of information. In addition, weights are only initialized by the source nodes, so no other node can bring to the flow.

4 Ascending flow in citation networks

We provide now a base measure called *ascending flow* and discuss its complexity. We then extend it to several variants, such as one that is restricted in depth, hence that fits better a dynamic context. Two natural definitions help defining our framework and its integration with existing metrics.

Definition 4.1 (Related). Two articles a and b are said to be related if and only if there exist a path from a to b or from b to a . They are k -related if they are related and if the shortest path between them is at most of length k .

Definition 4.2 (k -diffuse). A measure of a node v is k -diffuse when it limits its computation to a subgraph composed of the k -related nodes of v

4.1 Ascending flow

We can now model the stream of knowledge as a flow in our citation network. Indeed, each node — being a publication — produces some information and this production of information gives credit to their ancestors (in history, or successors in the DAG) as they refer to them. This translates into the framework as:

$$F(\mathcal{N}(v)) = \sum K(k_i)/d^+(k_i) + \alpha_v \quad (5)$$

Where α_v represents the information created by the publication v — in practice we set $\alpha_v = 1$. Hence, the more a publication is influential the more credit it will propagate to its ancestors. In contrast to the previous *Flow metric*, our ascendant flow is not only applied to the reversed DAG, but is also equivalent to the sum of the flows computed for each sub-DAG induced by each node.

The ascending flow, formalized above, can be implemented as algorithm 4. It is important to notice that each arc is visited only once and that the total number of visits of all nodes is also equal to the number of arcs. The time complexity of our algorithm is then $\Theta(m)$ where m is the number of arcs. This key property is inherent to the pseudo-DAG nature of our citation network. As described in section 3, citation networks can be converted to DAG with minimum loss of information. However, even a linear time complexity is often too costly for large dynamic network.

Algorithm 4 ascending flow

Input: A citation network with nodes (articles) and arcs (citations)

An empty dequeue Q (FIFO)

Output: The ascending flow on each node (article) and each arc (citation)

- 1: Initialize each article v with flow value $\alpha_v = 1$
 - 2: Color each arc in white
 - 3: Add all leaves in Q
 - 4: **while** Q is not empty **do**
 - 5: $v \leftarrow pop_first(Q)$
 - 6: **for** each w son of v **do**
 - 7: Color each (v, w) in blue
 - 8: $\alpha_w \leftarrow \alpha_w + \alpha_v/d^-(v)$
 - 9: **if** all incoming arcs of w are blue **then**
 - 10: $Q \leftarrow push_last(w)$
 - 11: **end if**
 - 12: **end for**
 - 13: **end while**
-

4.2 Depth restriction and dynamic graph

As discussed above, one issue of computing the ascending flow of a node v from our definition is that it needs the computation of all successors own influence. Such a constraint is expansive in the context of a dynamic network, for instance citation networks — in the case of citation network, publication are usually added, not removed. To adapt our previous algorithm, we first need to introduce an update function starting from a single leaf (a new publication). We consider the network initializes as in algorithm 4 but for the flow value on the nodes — that is kept between the updates. We then propagate upwards the flow value in all the subgraphs defined by the ancestors of this publication (Figure 1).

Recall the diffuse property in definition 4.2. Our base measures, the h -index and the number of citations, are respectively 2- and 1-diffuse by definition, whereas the ascending flow is ∞ -diffuse. In the real-world, we can consider that a publication that came a few generations after an original will relatively diverge from the original one, and would marginally contribute to the influence of the previous publication. The k -diffusion property can then take two forms: either we choose a generational limit k that cuts the added influence of nodes generated *after* k generations, or we can set an *evanescence* coefficient that progressively attenuates the contribution of a publication over its ancestors. In the case of a dynamic citation network, a k -diffuse measure is very quick to compute when k is a small constant as in Figure 2b.

This depth parameter additionally allows us to reconnect with known measures. For example, the h -index is 2-diffuse and it would not make sense to extend its definition. In turn, the number of citations — which is also the in-degree ($d^-(v)$) — is 1-diffuse. This can then be easily translated in a k -diffuse measure, the k -degree, which would be the number of publications created until generation k . Then, an ∞ -degree would be the number of all publications seeded by v even indirectly.

5 Experimental results

We now study our framework on a real-world setting. We used an available citation graph from 2003 KDD Cup: Arxiv HEP-TH[15]¹. It consists in an archive of 27,770 publications with 352,807 (internal) citations from the well-known ArXiv website of pre-prints in the domain of high energy physics theory, archived between January 1993 to April 2003. The resulting graph (Figure 2a) is not acyclic due to the nature of publications in ArXiv — some publications have been updated with cross-references to others. We can however consider this graph as pseudo-acyclic because number and size of the cycles are limited (a few cycles of size 2 and 1 cycle of size 3). In our setting we simply remove those edges to keep the properties of a DAG. A resulting excerpt of the graph is shown in Figure 2c.

As we have defined the generalized version of the number of citations in our framework and the h -index, we compare these measures altogether. We compare the Pearson and Spearman correlation coefficients of these measures together with the

¹ available at: <http://snap.stanford.edu/data/cit-HepTh.html>

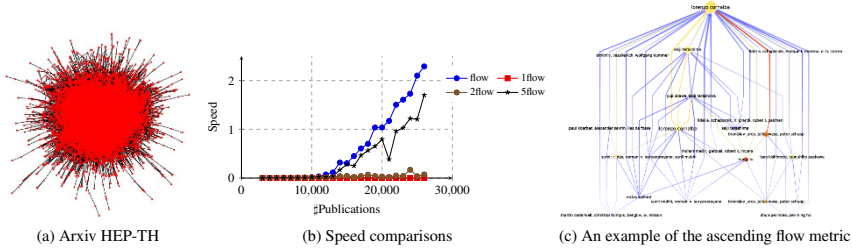


Fig. 2: (a) The main connected component of the ArXiv HEP-TH (high energy physics theory) citation network with 27770 nodes (articles) and 352807 arcs (citations). (b) Speed comparisons of our algorithm in case of k -diffuse limitations. (c) An example of the ascending flow metric in an excerpt of 22 nodes (60 edges) of our dataset, rooted by a publication by Lorenzo Cornalba. The size of nodes corresponds to their ascending flow in this subgraph. The color of nodes and edges (from blue to red) is actually their ascending flow in the real global dataset — we can see that Hong Liu’s publication has probably been a seed for more knowledge than of its ancestor Lorenzo Cornalba. flows

Pearson	Spearman													
	h -index	ascending flow	∞ -degree	1-degree	2-degree	5-degree	10-degree	20-degree	1-flow	2-flow	5-flow	10-flow	20-flow	
h -index	-	0.821	0.765	0.958	0.954	0.849	0.770	0.765	0.776	0.809	0.807	0.807	0.807	
ascending flow	0.546	-	0.758	0.858	0.807	0.764	0.759	0.758	0.961	0.990	0.991	0.991	0.991	
∞ -degree	0.476	0.267	-	0.715	0.809	0.947	1.000	1.000	0.654	0.710	0.714	0.714	0.714	
1-degree	0.768	0.648	0.265	-	0.920	0.794	0.719	0.715	0.856	0.863	0.860	0.860	0.860	
2-degree	0.850	0.670	0.375	0.766	-	0.908	0.815	0.809	0.725	0.776	0.775	0.775	0.775	
5-degree	0.626	0.347	0.856	0.367	0.546	-	0.952	0.947	0.657	0.714	0.716	0.716	0.716	
10-degree	0.483	0.270	0.999	0.268	0.381	0.865	-	1.000	0.654	0.710	0.714	0.714	0.714	
20-degree	0.476	0.267	1.000	0.265	0.375	0.856	0.999	-	0.654	0.710	0.714	0.714	0.714	
1-flow	0.637	0.694	0.330	0.904	0.638	0.367	0.332	0.330	-	0.987	0.985	0.985	0.985	
2-flow	0.664	0.814	0.337	0.892	0.712	0.390	0.339	0.337	0.969	-	1.000	1.000	1.000	
5-flow	0.656	0.823	0.341	0.879	0.704	0.392	0.344	0.341	0.964	0.999	-	1.000	1.000	
10-flow	0.656	0.823	0.341	0.879	0.704	0.392	0.344	0.341	0.964	0.999	1.000	-	1.000	
20-flow	0.656	0.823	0.341	0.879	0.704	0.392	0.344	0.341	0.964	0.999	1.000	1.000	-	

Table 1: Comparison of Pearson coefficients (bottom left, correlation of values) and Spearman coefficient (top right, correlation of ranks) between all measures.

following assumption: if the ascendant flow can reconnect at least partially to the notion of degree and h -index, we can then validate the relevance of our framework. Results of the analysis are presented in Table 1 and Figure 3.

First, when comparing the h -index, the number of citations, and the total number of publications produced by a work, we can notice a clear difference on our four basic metrics: the number of citations ($=1$ -degree), the number of publications generated ($=\infty$ -degree), the h -index and the ascendant flow. We additionally varied the depth of degree and flow in $\{1, 2, 5, 10, 20, \infty\}$. A second observation is that the limitation in depth of our measure is consistent with what we observe when limiting the depth

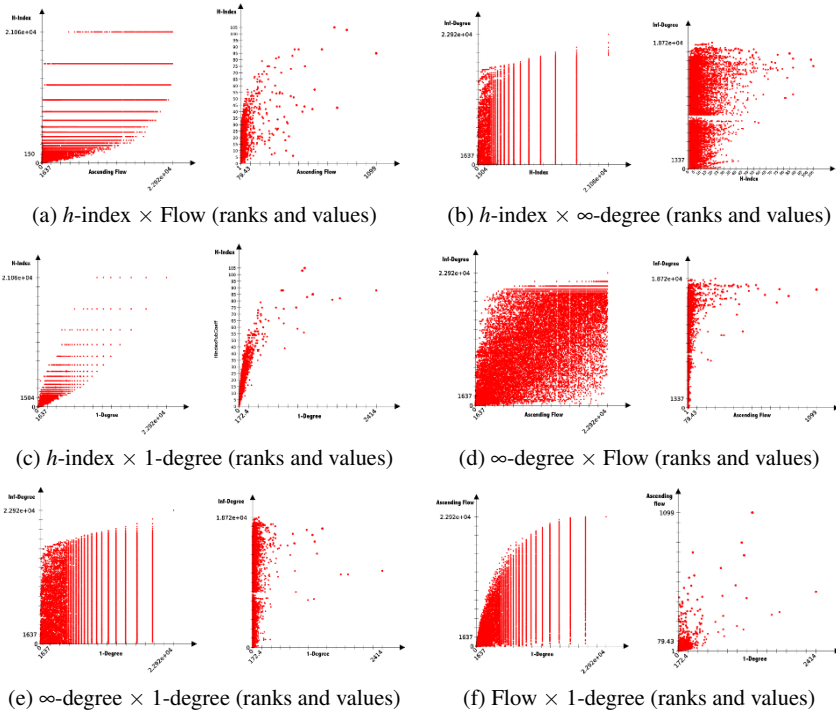


Fig. 3: Comparative distribution of ranks and values among 1-degree (*i.e.* number of citations of a publications, ∞ -degree (*i.e.* number total of generated publications), h -index, and ascendant flow. The plots well illustrate the difference between what those statistics are measuring.

of the k -degree (the most correlated i -flow for a j -degree is when $i = j$), and the higher k for the k degree, the more it diverges from the k -flow.

Our main observation, is, by value, the h -index is most correlated to the 2-degree. This makes complete sense, since the h -index is also limited in depth at 2 for which it considers a subset of publications. In contrast, when it comes to rankings, the h -index is most correlated to the 1-degree which is equivalent to the number of citations. Interestingly, our ascending flow also shares most correlations with the 2-degree as well and ranks with the 1-degree. This interesting effect may also be observed in Figure 2c showing that most publications bringing influence to the source publication has done it already in depth two. The link between the h -index and the degree is further observable in Figure 3.

In terms of computation, from $k = 2$, the ranks obtained by the k -flow are .99 similar of those of the regular flow so when a gain of computation is needed, one can use k -diffuse version of the algorithm (Figure 2b).

Now we can compare publications of a same h -index and published around the same date which have very different flow measures. We took 2 publications with very different ascending flows: the first one shows a flow at 11.23 (Figure 4a, left),

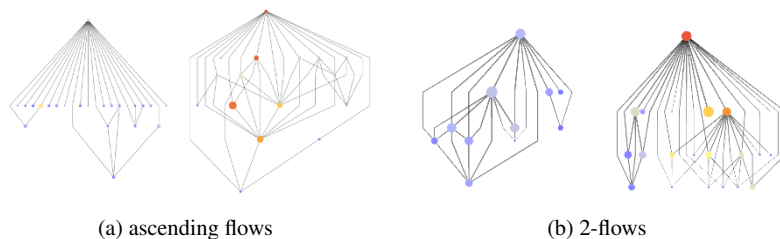


Fig. 4: Comparison of direct citations of four publications with h -index =6. The top node is one original publication, and all other nodes its citing nodes (a) Comparison of the general ascending flows with two extreme values: left ID920426 (flow=425.4), right ID9201019 (flow=11.2) (b) Comparison of 2-flows with two extreme values: left ID9201079 (2-flow=2.3), right ID9201058 (2-flow=21.6). Relative node size (between couples of pictures) correspond to h -index values for each node. Node color correspond to, (a) ascending flow, (b) 2-flow.

while the second one displays a flow measure at 425.44 (Figure 4a, right). Their in-degree does not vary that much (21 vs. 16 for the most influential), however, the 2-degree makes the difference (151, vs. 707). That means in average, the publications citing the most influential work produce more than four times more citations in turn – average h -index is 3.2 vs. 10.6. Note also that our measure takes into account how the information is spread out. In the first case, we have 390 citing edges out, while we have 171 in the other case.

We repeated the same experiment with two varying 2-flow measures (h -index =6 and similar date of publication): the first one is 2.25 with 10 citations (Figure 4b, left), and the second one is 21.59 with 20 citation (Figure 4b, right). The average h -index in the least influential one is actually higher (3.45) than of the most influential (1.80). However, the most influential has seeded 102 citations (2-degree) vs. 17 edges outs, when the first one 68 citations for 182 citing out. The flow measures then capture much more details of the graph of produced by citations than the h -index allows.

6 Discussion and conclusion

We have shown that the production and diffusion of knowledge can be modeled in a recursive framework that studies flows in DAGs, with a natural interpretation of the notion stream of knowledge. The framework allows for other known metrics to be embedded, and for efficient computation on large dynamic graphs. We applied our different flows and compared them with other known measures. By comparing the ascendant flow with the h -index we clearly see a correlation. The h -index has been a very popular indicator and useful for predictions and scientometrics. Our measure's interpretation is straightforward, and this correlation goes in favor of the relevance of the h -index. But we do not fully correlate with the h -index, and many cases that are oversimplified by the h -index can be finer described by the ascending flow.

We looked for differences in flow when the h -index gives a same value. We found cases with large differences, and explain the differences as follows: the h -index gives a rough estimation of a publication's production of knowledge, but it does not take into account how each citation refer to the original work. The flow measure, even 2-diffuse, is reinforced by two factors. A first one is something similar to a "community" effect in citations, *i.e.* when the citations produced also cite each other in relative proportion, in comparison to citations "outside" that "community" of citations. For example, this happens when a paper has an influence in developing a community of research, the large the community, the greater the flow. The second effect gets more relevant as the depth of diffusion is greater. It is somewhat close to the hubs and authorities effect: the more citations a paper gets from influential papers the more influential it will get.

The interpretation of flow we propose is much more flexible than the h -index, and can fairly support a wide range of parameters for scientists to conduct further experiments (such as additional weights, edge filtering, depth of influence, *etc.*). More than a metric, when studying the influence of a work (or a collection of works), we argue that the structure of the flow of knowledge it produces, *i.e.* the DAG generated by a publication and its citations should be taken into account.

Although our study does not hold for an evaluation for which a comparison with many other metrics and regression would have been necessary, we still have set and validated the basis of our framework – in that it comprises well other known measures. Now, this will allow to take our graphs to another level of complexity – namely multiplex DAGs. H -index would apply with difficulty in a multiplex network, but we are currently focusing our effort in studying the ascendant flow in a version of our citation graphs where different routes could be considered in parallel (because knowledge does not flow equally in all citation sources). Among our future work is also the application to the analysis of news documents. Indeed, DAGs also apply to the study of closely related documents – even if there is no citation relationship, the time dependency between closely related documents can maintain the DAG assumption. Extending our study to other databases, such as DBLP, we would like to conduct case studies on authors and journals this time, to observe the influence of Nobel prizes or high standard journals.

References

- [1] Alkemade, F., Castaldi, C.: Strategies for the diffusion of innovations on social networks. *Computational Economics* **25**(1-2), 3–23 (2005)
- [2] Assad, A.: Multicommodity network flows a survey. *Networks* **8**(1), 37–91 (1978)
- [3] Auber, D.: Using strahler numbers for real time visual exploration of huge graphs. In: *International Conference on Computer Vision and Graphics*, vol. 1, p. 3 (2002)
- [4] B., V.: Efficient algorithms for citation network analysis. *CoRR* **cs.DL/0309023** (2003)
- [5] Bornmann, L., Daniel, H.: What do citation counts measure? a review of studies on citing behavior. *Journal of Documentation* **64**(1), 45–80 (2008)
- [6] Bucur, O., Almasan, A., Zubarev, R., et al.: An updated h -index measures both the primary and total scientific output of a researcher. *Discoveries* **3**(3) (2015)

- [7] Cattuto, C., Quaghiotto, M., et al.: Time-varying social networks in a graph database: A neo4j use case. In: First Int. Workshop on Graph Data Management Experiences and Systems, GRADES '13, pp. 11:1–11:6. ACM (2013)
- [8] Chen, P., Redner, S.: Community structure of the physical review citation network. *Journal of Informetrics* **4**(3), 278–290 (2010)
- [9] Cointet, J., Roth, C.: How realistic should knowledge diffusion models be? *Journal of Artificial Societies and Social Simulation* **10**(3), 5 (2007)
- [10] Cowan, R., Jonard, N.: Knowledge creation, knowledge diffusion and network structure. In: *Economics with heterogeneous interacting agents*, pp. 327–343. Springer (2001)
- [11] Cowan, R., Jonard, N.: Network structure and the diffusion of knowledge. *Journal of economic Dynamics and Control* **28**(8), 1557–1575 (2004)
- [12] Delest, M., Don, A., Benois-Pineau, J.: Dag-based visual interfaces for navigation in indexed video content. *Multimedia Tools and Applications* **31**(1), 51–72 (2006)
- [13] Egghe, L.: Theory and practise of the g-index. *Scientometrics* **69**(1), 131–152 (2006)
- [14] Ernst, D., Kim, L.: Global production networks, knowledge diffusion, and local capability formation. *Research policy* **31**(8), 1417–1429 (2002)
- [15] Gehrke, J., Ginsparg, P., Kleinberg, J.: Overview of the 2003 kdd cup. *ACM SIGKDD Explorations Newsletter* **5**(2), 149–151 (2003)
- [16] Gibbons, M., Johnston, R.: The roles of science in technological innovation. *Research Policy* **3**(3), 220–242 (1974)
- [17] Gibbons, M., Limoges, C., Nowotny, H., Schwartzman, S., Scott, P., Trow, M.: *The new production of knowledge: The dynamics of science and research in contemporary societies*. Sage (1994)
- [18] Herman, I., Marshall, M.S., Melançon, G., et al.: Skeletal Images as Visual Cues in Graph Visualization, pp. 13–22. Springer Vienna (1999)
- [19] Hirsch, J.E.: An index to quantify an individual's scientific research output **102**(46), 16,569–16,572 (2005)
- [20] Hirsch, J.E.: An index to quantify an individual's scientific research output that takes into account the effect of multiple coauthorship. *Scientometrics* **85**(3), 741–754 (2010)
- [21] Hummon, N., Dereian, P.: Connectivity in a citation network: The development of dna theory. *Social networks* **11**(1), 39–63 (1989)
- [22] Liu, J., Lu, L.: An integrated approach for main path analysis: Development of the hirsch index as an example. *Journal of the American Society for Information Science and Technology* **63**(3), 528–542 (2012)
- [23] Mueller, M., Bogner, K., Buchmann, T., et al.: Simulating knowledge diffusion in four structurally distinct networks: An agent-based simulation model (2015)
- [24] Pendlebury, D.A.: The use and misuse of journal metrics and other citation indicators. *Archivum immunologiae et therapiae experimentalis* **57**(1), 1–11 (2009)
- [25] Reuters, T.: The thomson reuters impact factor. thomson-reuters.com/products_services/science/free/essays/impact_factor/ (2012)
- [26] Strahler, A.N.: Quantitative analysis of watershed geomorphology. *Eos, Transactions American Geophysical Union* **38**(6), 913–920 (1957)
- [27] Van Raan, A.F.: Measuring science. In: *Handbook of quantitative science and technology research*, pp. 19–50. Springer (2004)
- [28] Waltman, L.: A review of the literature on citation impact indicators. *Journal of Informetrics* **10**(2), 365 – 391 (2016)
- [29] Wuchty, S., Jones, B.F., Uzzi, B.: The increasing dominance of teams in production of knowledge. *Science* **316**(5827), 1036–1039 (2007)

Detecting Nestedness in Graphs

Alexander Grimm and Claudio J. Tessone

Abstract Many real-world networks have a nested structure. Examples range from biological ecosystems (e.g. mutualistic networks), industry systems (e.g. New York garment industry) to inter-bank networks (e.g. Fedwire bank network). A nested network has a graph topology such that a vertex's neighborhood contains the neighborhood of vertices of lower degree. Thus –upon node reordering– the adjacency matrix is stepwise, and it can be found in both bipartite and non-bipartite networks. Despite the strict mathematical characterization and their common occurrence, it is not easy to detect nested graphs unequivocally. Among others, there exist three methods for detection and quantification of nestedness that are widely used: BINMATNEST, NODF, and FCM. However, these methods fail in detecting nestedness for graphs with low (NODF) and high (NODF, BINMATNEST) density or are developed for bipartite networks (FCM). Another common shortcoming of these approaches is the underlying assumption that all vertices belong to a nested component. However, many real-world networks have solely a sub-component (i.e. not all vertices) that is nested. Thus, unveiling which vertices pertain to the nested component is an important research question, unaddressed by the methods available so far. In this contribution, we study in detail the algorithm *Nestedness detection based on Local Neighborhood* (NESTLON) [7]. This algorithm detects nestedness on a broad range of nested graphs independently of their density and resorts solely on local information. Further, by means of a benchmarking model we are able to tune the degree of nestedness in a controlled manner and study its efficiency. Our results show that NESTLON outperforms both BINMATNEST and NODF.

Alexander Grimm (e-mail: alexander.grimm@business.uzh.ch)✉ · Claudio J. Tessone (e-mail: claudio.tessone@business.uzh.ch)✉
URPP Social Networks, Department of Business Administration, University of Zurich

1 Introduction

Two vertices are nested if the neighborhood of the one with larger degree contains the neighborhood of the lower degree one. We call *nested component* of a graph the maximum set of vertices that are nested. Following, a graph is nested if the extent of the nested component is such that it embraces all vertices. This definition applies in both bipartite and non-bipartite networks. Nested graphs include some common topologies like fully-connected ones or stars. In real-world networks, some edges violate the definition of pairwise nestedness given above; in this case, the lower the number of these violations, the larger the degree of nestedness of the network.

In Ecology, as it was discovered in the last decade, mutualistic networks show a pronounced degree of nestedness [4]. In Economics, e.g. the New York garment industry including 10'000 manufacturers over a period of 18 years was found to exhibit this property as well [15]. Among non-bipartite networks there are several examples of networks that show large degrees of nestedness: like inter-bank networks [13], and trade relations between countries [9].

Four methods have gained particular attention for detecting and quantifying Nestedness in the last decade: *Binary matrix nestedness temperature calculator* (BINMATNEST) [11], based on *Nestedness Temperature Calculator* (NTC) [2], *Nestedness metric based on overlap and decreasing filling* (NODF) [1], and *Fitness-Complexity Metric* (FCM) [14]. Nonetheless, these methods detect nestedness for only a specific density range (BINMATNEST, NTC and NODF fail in detecting nestedness for high density networks) or a specific class of graphs (FCM was developed for only bipartite ones).

All four methods assume that all vertices belong to a single nested component but, in general, this is not necessarily true. Such component might include solely a subset of vertices while the others lay outside it. Therefore, it is an important research question to devise a method that identifies the individual vertices that belong to a nested component. This question remains unaddressed by the methods available so far.

The widely used BINMATNEST is based on NTC, which compares the focal adjacency matrix with a "perfect ordered" matrix. The less these two matrices deviate from each other, the more the graph is judged as nested. However, the matrix of "perfect order" is a normative concept characterized by a static isocline [2] (i.e. matrix is filled up to the secondary diagonal). Both methods judge graphs only as nested if they have this particular "perfect order". They fail in detecting graphs that have locally nested components. This static and normative concept of nestedness relies only on global information (i.e. irrespective of local neighborhoods in the nested components). For large datasets it is important to develop methods for detecting nestedness that rely solely on local information, because they scale better [7].

In this contribution we review the method *Nestedness detection based on Local Neighborhood* (NESTLON) that reliably detects nestedness irrespective of graph density and network type (i.e. bipartite and non-bipartite networks) [7]. Although in this contribution we focus on non-bipartite graphs (for the sake of simplicity), all the results are easily extensible to bipartite ones.

The remainder of the paper is organized as follows. In the next section section we provide an overview about nestedness in graphs and the current methods for detecting it. In "Algorithm" section we review the alternative method NESTLON for detecting nestedness. In "Robustness Analysis" section we compare commonly used algorithms with NESTLON on a benchmarking graph. The final section concludes and discussed the main contributions of this Paper.

2 The Notion of Nestedness

2.1 Definition of Nestedness

We first give a colloquial definition of nestedness and later a proper mathematical definition. In a nested graph the neighborhood of a vertex includes the neighborhoods of vertices which have lower degrees ¹. Therefore, by sorting the adjacency matrix of a nested graph by degree (i.e. the number of direct neighbors) we obtain a stepwise matrix. For example, a star is nested and has a stepwise matrix. A star's central vertex has the highest degree (i.e. this vertex is connected every other vertex) and all other vertices have degree one (i.e. they are all connected only to the central high degree vertex) while the neighborhoods of all lower degree vertices are included in the neighborhood of the high degree vertex. Therefore, the adjacency matrix of a star has just one large step (i.e. from maximum degree to one-degree).

For a proper mathematical characterization we briefly recapture the nomenclature for graphs. The adjacency matrix, A , characterizes the topology of a graph object G . An non-zero entry in the adjacency matrix, $a_{ij} \neq 0$, indicates an edge between the two vertices i and j . Each vertex has a degree, k_i , which is the number of neighbors it is connected to. The total number of edges is e and the total number of vertices is n . N is the set of all vertices and E is the set of all edges. A graph can be decomposed by the concept of degree partition [10]:

Definition 2.1. Let $G = (N, E)$ be a graph whose distinct positive degrees are $k_{(1)} < k_{(2)} < \dots < k_{(m)}$ and let $k_{(0)} = 0$ (even if no vertex with degree 0 exists in G). Further, define $\mathcal{D}_i = \{v \in N : k_v = k_{(i)}\}$ for $i = 0, \dots, m$. Then the set-valued vector $\mathcal{D} = (\mathcal{D}_0, \mathcal{D}_1, \dots, \mathcal{D}_m)$ is called the degree partition of G .

With this concept of degree partition a nested graph can be expressed as follows [10]:

Definition 2.2. Consider a nested graph $G = (N, E)$ and let $\mathcal{D} = (\mathcal{D}_0, \mathcal{D}_1, \dots, \mathcal{D}_m)$ be its degree partition. Then the vertices N can be partitioned in independent sets \mathcal{D}_i , $i = 1, \dots, \lfloor m/2 \rfloor$, and a dominating set $\bigcup_{i=\lfloor m/2 \rfloor+1}^m \mathcal{D}_i$ in the graph $G' = (N \setminus \mathcal{D}_0, E)$. Moreover, the neighborhoods of the vertices are nested. In particular, for each vertex $v \in \mathcal{D}_i$, $i = 1, \dots, m$, we obtain the sets of vertices as

$$N_v = \begin{cases} \bigcup_{j=1}^i \mathcal{D}_{m+1-j} & \text{if } i = 1, \dots, \lfloor m/2 \rfloor; \\ \bigcup_{j=1}^i \mathcal{D}_{m+1-j} \setminus \{v\} & \text{if } i = \lfloor m/2 \rfloor + 1, \dots, m. \end{cases} \quad (1)$$

¹ This definition is for non-bipartite graphs, for bipartite graphs a similar definition holds [4].

An adjacency matrix is stepwise if the following definition holds [5]:

Definition 2.3. A stepwise matrix A is a symmetric, binary $(n \times n)$ matrix with elements a_{ij} satisfying the following condition: if $i < j$ and $a_{ij} = 1$, then $a_{hk} = 1$ whenever $h < k \leq j$ and $h \leq i$.

Thus, a nested graph has a stepwise adjacency matrix and its degree partition can be separated into an independent and a dominating sets.

A measure for determining the filling of an undirected graph is the density.

Definition 2.4. The density of an undirected graph is given by

$$\gamma_d = \frac{2 \cdot e}{n \cdot (n-1)} \quad (2)$$

In the following we propose a measure for counting the number of holes in a graph. We compare the neighborhoods of two vertices i and j . If the lower degree vertex j has a neighbor l , which is not neighbor of i , we will count a hole (because it appears as such in the sorted adjacency matrix). From there, the density of holes can be computed [7]

Definition 2.5. The total number of holes in an unweighted graph is given by

$$\gamma_h = \frac{\sum_{i,j \in N} \Theta(k_i - k_j) \sum_{l \in N} (1 - a_{li}) \cdot a_{lj}}{\sum_{i,j \in N} \Theta(k_i - k_j) \min(n - k_i, k_j)} \quad (3)$$

with $\Theta(x)$ the Heaviside function:

$$\Theta(x) = \begin{cases} 0 & \text{if } x < 0; \\ \frac{1}{2} & \text{if } x = 0; \\ 1 & \text{if } x > 0. \end{cases}$$

2.2 Detecting and Measuring Nestedness

In this section we briefly discuss three commonly used methods for quantifying nestedness in graphs. These measures are BINMATNEST (based on the NTC), NODF, and FCM.

Binary matrix nestedness temperature calculator (BINMATNEST)

NTC performs insufficiently if the number of holes in a graph is high. Therefore, BINMATNEST uses a genetic algorithm that reorders rows and columns so that the packing of the matrix increases. The matrix temperature T is a measure of how equally the edges are distributed across the matrix. If all edges are in the upper left corner the temperature is minimal ($T \rightarrow 0$). If all edges are equally distributed in the matrix the temperature is maximal ($T \rightarrow 100$). The normalized temperature of the adjacency matrix is given by the following expression [6]:

$$\mu_{BIN} = \frac{100 - T}{100} \quad (4)$$

If $\mu_{BIN} = 1$ (0) the matrix temperature will be minimal $T = 0$ (resp. maximal $T = 100$).

Nestedness metric based on overlap and decreasing filling (NODF)

NODF was developed for bipartite networks of ecological systems [1] but it is applicable to square matrices, too. This method is independent of row and column order since it computes the paired nested degree for each pair of both columns and rows. However, in contrast to BINMATNEST this method does not reshuffle the matrix. For the whole matrix the sum of nestedness degrees of all paired rows and columns is the total nestedness normalized by the number of all pairs. The NODF metric assigns a value M_{ij}^H to each neighboring pair of vertices ij :

$$M_{ij}^H = \begin{cases} 0, & \text{if } k_i = k_j \\ \frac{n_{ij}}{\min(k_i, k_j)}, & \text{otherwise} \end{cases} \tag{5}$$

The total number of common edges among the two vertices i and j is given by n_{ij} . The procedure is carried out for rows (M_{ij}^R) and columns (M_{ij}^A) analogously. Finally, the total nestedness for square matrices is then given by [12]:

$$\mu_{NODF} = \frac{\sum_{i<j}^P M_{ij} + \sum_{i<j}^A M_{ij}}{\frac{2 \cdot n(n-1)}{n}} \tag{6}$$

An advantage of NODF is its independence of matrix shape because it goes through both rows and columns [12]. However, this method fails in detecting nestedness for nested graphs of low and high density because it cancels out all terms for vertices of same degree.

Fitness-Complexity Metric (FCM)

FCM ranks vertices in an iterative and non-linear process [14]. The iteration process couples a fitness term to a complexity term. Since FCM was solely developed for bipartite networks, we will not use it as a benchmark in this contribution.

2.3 Benchmark Graphs

We require a solid benchmarking framework for comparing robustness and reliability among different nestedness detection methods. A benchmark graph needs to differ in its network characteristics (i.e. degree distribution, graph density, vertex centrality, etc.) but keep a certain level of nestedness. The authors of [8, 9] propose a coherent formation process for generating nested graphs with a single exogenous parameter α that influences the topology of the generated graphs fundamentally. This network formation process has two contrasting dynamics, edge creation and severance. First, the edge creating dynamics allows each vertex to create an edge to the most central vertex in its second-order neighborhood (i.e. the neighbors of its own neighbors) with a probability α . Second, each vertex may severe the edge to the least central neighbor in its first-order neighborhood with the complementary probability $1 - \alpha$. By changing α we can tune a nested graph between two limiting cases. On the one hand, we obtain a star topology for $\alpha \rightarrow 0$ and, on the other hand, we obtain a

fully-connected graph for $\alpha \rightarrow 1$. A first-order phase transition exists at the critical value $\alpha = 1/2$ [8].

The degree partition for the independent set of the nested graph is given by the following definition [9]:

Definition 2.6. For $0 < \alpha \leq 1/2$ and $n \rightarrow \infty$ the asymptotic expected proportion of vertices n_k in the independent set with degrees $k = 0, 1, \dots, k^*$ is given by

$$n_k = \frac{1 - 2\alpha}{1 - \alpha} \left(\frac{\alpha}{1 - \alpha} \right)^k \quad (7)$$

where

$$k^*(n, \alpha) = \frac{\ln \left(\frac{(1-2\alpha)n}{2(1-\alpha)} \right)}{\ln \left(\frac{1-\alpha}{\alpha} \right)} \quad (8)$$

In this contribution we utilize this network topology to create benchmark graphs. In addition, it is possible to weaken the perfectly nested topology by an incremental increase of random rewiring of edges. This process works as follows. First, for a randomly chosen vertex we determine all of its next neighbors. Second, a connection to a randomly chosen neighbor is cut and the focal vertex is connected to another vertex to which it previously was not connected to. If a vertex is isolated or is connected to all nodes in the network, nothing happens. The total number of rewired edges e_{new} is given by the parameter ρ_{rew} . These two quantities are linked in the following way: $e_{new} = \rho_{rew} \cdot n$. The higher ρ_{rew} the more edges get randomly rewired. This process can be seen as a simplification of other rewiring mechanisms in nested networks [3].

3 Algorithm

In this section we briefly review the algorithm NESTLON as a method for detecting a nested component in graphs and its constituents [7]. The simple main concept behind the algorithm is to follow the definition of nestedness closely. NESTLON judges whether the neighborhood of a vertex includes the neighborhood of lower degree vertices in an iterative manner. A vertex belongs to the nested component if it respects the local definition of nestedness to an acceptable degree.

The method iterates through the connected component of a graph starting with the highest degree vertex and, therefore, is applicable on both bipartite and non-bipartite graphs. The procedure is analogous for either in-degree or out-degree (for simplicity we refer to the term degree in the following). We use the algorithm on a graph that is sorted by degree centrality. The algorithm performs the following steps subsequently:

Algorithm: *Nestedness detection based on Local Neighborhood (NESTLON)*

Conventions:

- n Number of vertices in the graph.
- k_i Degree of vertex i .
- $\mathcal{N}_i^{(1)}$ First-order neighborhood of vertex i .
- $\mathcal{N}_i^{(1+)}$ Extended first-order neighborhood of vertex i ($\mathcal{N}_i^{(1)} \cup \{i\}$).
- ζ_i Number of positive confirmations that the neighborhood of vertex i includes the neighborhoods of its first-order neighbors.
- Λ List of candidates (i.e. vertices that might belong to nested component).
- $|\cdot|$ Number of elements in a set.

Input:

- \mathbf{A} Adjacency matrix of the graph object.
- θ_{con} Confirmation parameter of neighborhood similarity-
- θ_{nest} Parameter for counting focal vertex to nested component.

Output:

- V_{nest} Elements of nested component (i.e. vertices that belong to nested component).

Algorithm NESTLON

```

1:  $V_{nest} \leftarrow \{\}$ 
2:  $\Lambda \leftarrow \{i^*; i^*/k_{i^*} = \max(k_i)\}$ 
3: while  $\Lambda \neq \emptyset$  do
4:   for  $i \in \Lambda$  do
5:      $\zeta_i \leftarrow 0$ 
6:     for  $j \in \mathcal{N}_i^{(1)}$  do
7:       if  $\frac{|\mathcal{N}_j^{(1+)} \cap \mathcal{N}_i^{(1+)}|}{\min(|\mathcal{N}_j^{(1+)}|, |\mathcal{N}_i^{(1+)}|)} > \theta_{con}$  then
8:          $\zeta_i \leftarrow \zeta_i + 1$ 
9:          $\Lambda \leftarrow \Lambda \cup \{j\}$ 
10:      end if
11:    end for
12:    if  $\frac{\zeta_i}{|\mathcal{N}_i^{(2)}|} > \theta_{nest}$  then
13:       $V_{nest} \leftarrow V_{nest} \cup \{i\}$ 
14:    end if
15:  end for
16: end while

```

The outcome of the algorithm is a set of vertices that belong to the nested component V_{nest} . Dividing the number of nested vertices by the highest degree of the graph is then a measure of the size of the component:

$$\mu_{NEST} = \frac{|V_{nest}|}{\max(k_i)}, \text{ with } i \in N \quad (9)$$

This method has several important features. It is independent on the adjacency matrix shape and size. In contrast to NODF it calculates nestedness for rows and columns independently. Compared to NODF and BINMATNEST it can detect nested graphs irrespective of their density. We will investigate the robustness of the algorithm in the next section.

4 Robustness Analysis

A robust algorithm can detect the nested component independently of degree distribution, graph density, matrix shape and matrix size. Such a robust algorithm should identify all vertices that fulfill the criterion of nested neighborhoods (i.e. a higher degree vertex includes the neighborhood of a lower degree vertex). Therefore, we can evaluate an algorithm's robustness on such a benchmark graphs, in which all vertices belong to a single nested component. We create these graphs with the network formation process, which we already discussed in section "The Notion of Nestedness".

4.1 Calibration of NESTLON

Before we compare the values of robustness among the algorithms we need to calibrate the two exogenous parameters of the NESTLON algorithm (i.e. θ_{con} and θ_{nest}). The parameter θ_{con} is the confirmation threshold of neighborhood similarity and the parameter θ_{nest} is the threshold for counting a focal vertex to the nested component.

Calibration of NESTLON: Variation of θ_{con} and θ_{nest}

In fig. 4.1 we show the values of Nestedness for the NESTLON algorithm under variation of both parameters θ_{con} and θ_{nest} . The number of vertices the algorithm counts as nested does not differ for $\theta_{con} < 1$ but decreases for $\theta_{nest} \geq 0.5$. Because we deal with a perfectly nested graph (i.e. benchmark graph with $\alpha = 0.49$, $\rho_{rew} = 0$) both parameters shall be set so that NESTLON measures full nestedness (i.e. $\mu_{NEST} \stackrel{!}{=} 1$). Thus, we choose $\theta_{con} < 1$ and $\theta_{nest} < 0.5$ as reasonable for detecting nestedness.

Calibration of NESTLON: Adding Noise

In fig. 2 we show the NESTLON's ability in detecting the nested component on a benchmark graph with added noise (i.e. random rewiring of edges). In absence of rewiring (i.e. $\rho_{rew} = 0$) the algorithm includes all vertices as members of the nested component. For increasing random rewiring (i.e. $\rho_{rew} > 0$) the algorithm counts fewer vertices as part of the the nested component. This behavior is expected because the graph loses its nested structure with an increasing number of edge rewiring.

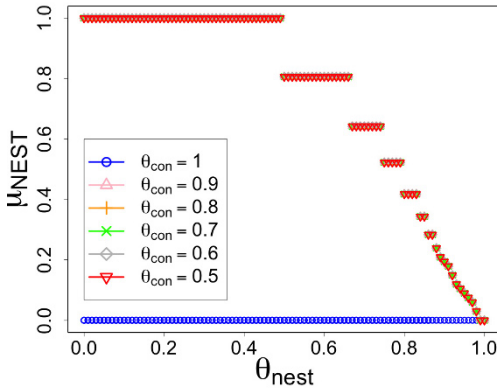


Fig. 1: Values of Nestedness for the NESTLON algorithm under variation of both exogenous parameters θ_{con} and θ_{nest} . We perform the computation on a benchmark graph of size $n = 500$ and $\alpha = 0.49$. Thus, all vertices belong to a single nested component. As we can see in the figure the thresholds are too rigid for $\theta_{con} = 1$ and $\theta_{nest} \geq 0.5$. Therefore, we choose $\theta_{con} < 1$ and $\theta_{nest} < 0.5$ as reasonable detection thresholds.

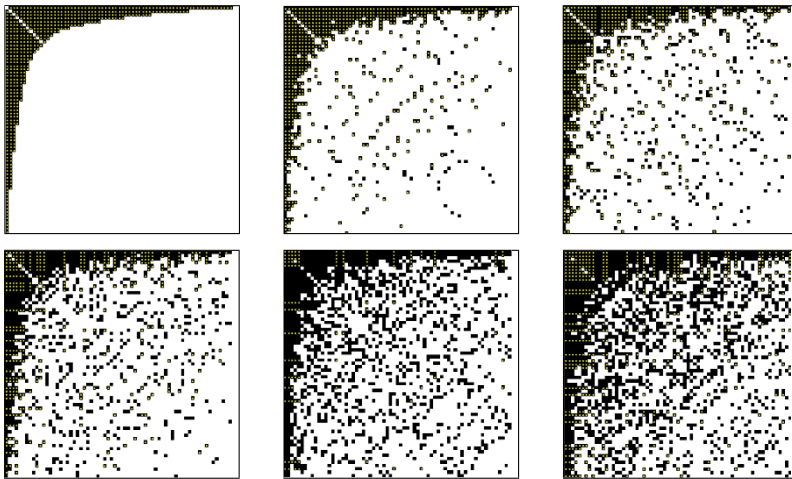


Fig. 2: Adjacency matrices of the benchmark graphs with additional noise: $\rho_{rew} = 0.0$ (top left), $\rho_{rew} = 1.0$ (top center), $\rho_{rew} = 2.0$ (top right), $\rho_{rew} = 3.0$ (bottom left), $\rho_{rew} = 5.0$ (bottom center), $\rho_{rew} = 7.0$ (bottom right). The vertices that are counted towards to the nested component by *NESTLON* are indicated by yellow dot.

Robustness Analysis: Filling Matrix

In fig. 3 we show the values of robustness measured among the three methods BINMATNEST, NODF and NESTLON on the benchmark graphs. By increasing α the matrix filling (i.e. network density γ_d) will increase, too. The benchmark graphs are nested by definition for every value of $\alpha \in [0, 1]$.

Although every benchmark graph is perfectly nested, BINMATNEST misses to detect all vertices as belonging to the nested component beyond the phase transition (i.e. $\alpha > 1/2$). For a fully connected network its genetic algorithm can not establish a

better packing by reordering rows and columns. NODF fails in detecting nestedness for graphs with low (i.e. $\alpha < 1/2$) and high density (i.e. $\alpha > 1/2$). Because this method cancels out all rows and columns of same degree it has a strong bias towards low nestedness for both low and high density graphs. However, NESTLON indicates an entirely nested network for every graph density (i.e. $\mu_{NEST} = 1$ for every value of $\alpha \in [0, 1]$).

Robustness Analysis: Adding Noise

In fig. 4 we compare the measured values of robustness among the three algorithms for increasing random rewiring ρ_{rew} . In absence of rewiring (i.e. $\rho_{rew} = 0$) the graph is still perfectly nested and, thus, we expect nestedness close to $\mu = 1$. For increasing rewiring (i.e. $\rho_{rew} > 0$) we expect that the nestedness decreases because the density of holes increases. BINMATNEST and NESTLON count all vertices to the nested component for $\rho_{rew} = 0$, whereas NODF recognizes only less than half of the vertices. By increasing noise NESTLON is significantly more parsimonious than the two other methods in judging vertices as nested. NODF has even a minimum at $\rho_{rew} \approx 4.5$. Beyond this minimum NODF detects a larger fraction of nested vertices although the graph increasingly converges to a random graph.

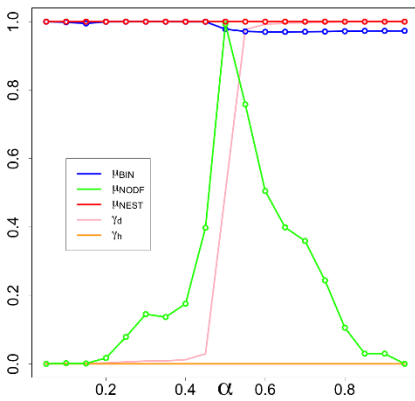


Fig. 3: Robustness in detecting the nested component among BINMATNEST, NODF and NESTLON on a benchmark graph. By definition all realizations of the benchmark graph are nested for all values of α . We perform the computation on a graph of size $n = 200$. The graph density (i.e. γ_d) increases with α , whereas the density of holes (i.e. γ_h) stays zero.

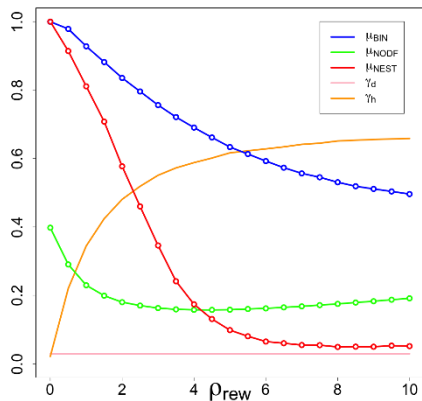


Fig. 4: Robustness in detecting the nested component among BINMATNEST, NODF and NESTLON on a benchmark graph with added noise. With increasing random rewiring ρ_{rew} the nested structure of the benchmark graph dissolves (i.e. increasing density of holes γ_h). We perform the computation on a graph of size $n = 200$ and with $\alpha = 0.45$ (i.e. $\gamma_d \approx 0.029$).

Conclusion

In this contribution we reviewed the novel method termed NESTLON for detecting a nested component in graphs. As shown, widely-used algorithms such as BINMATNEST and NODF compute unreasonable low values of nestedness on benchmark graphs with either low density (i.e. $\gamma_d < \frac{1}{2}$), NODF, or high density (i.e. $\gamma_d > \frac{1}{2}$), NODF and BINMATNEST. The method NESTLON overcomes these limitations and is applicable on both bipartite and non-bipartite graphs. The algorithm is purely based on the mathematical definition of nestedness and utilizes, thus, only local information. For the robustness analysis we created benchmark graphs with a network formation process. This network formation process allows us to tune the degree of nestedness in a controlled manner. In future work, we want to extend NESTLON to graphs with more than a single nested component.

Acknowledgements The authors acknowledge financial support from the University Research Priority Programme on Social Networks, University of Zurich.

References

- [1] Almeida-Neto, M., Guimarães, P., Guimarães, P.R., Ulrich, W.: A consistent metric for nestedness analysis in ecological systems: reconciling concept and measurement. *Oikos* **117**(March), 1227–1239 (2008). DOI 10.1111/j.2008.0030-1299.16644.x
- [2] Atmar, W., Patterson, B.D.: The Measure of Order and Disorder in the Distribution of Species in Fragmented Habitat. *International Association for Ecology* **96**(3), 373–382 (1993)
- [3] Bardoscia, M., Luca, G., Livan, G., Marsili, M., Tessone, C.J.: The Social Climbing Game. *Journal of Statistical Physics* **151**(3-4), 440–457 (2013). DOI 10.1007/s10955-013-0693-0
- [4] Bascompte, J., Jordano, P., Melián, C.J., Olesen, J.M.: The nested assembly of plant-animal mutualistic networks. *Proceedings of the National Academy of Sciences of the United States of America* **100**(16), 9383–9387 (2003). DOI 10.1073/pnas.1633576100
- [5] Brualdi, R., Hoffmann, A.: On the Spectral Radius of (0,1)-Matrices. *Linear Algebra and its Applications* **146**, 133–146 (1985)
- [6] Flores, C.O., Valverde, S., Weitz, J.S.: Multi-scale structure and geographic drivers of cross-infection within marine bacteria and phages. *The ISME Journal* **7**(3), 520–532 (2012). DOI 10.1038/ismej.2012.135. URL <http://dx.doi.org/10.1038/ismej.2012.135>
- [7] Grimm, A., Tessone, C.J.: Detecting the nested components of generic graphs (2017). In preparation
- [8] König, M.D., Tessone, C.J.: Network evolution based on centrality. *Physical Review E* **84**(5), 056108 (2011). DOI 10.1103/PhysRevE.84.056108
- [9] König, M.D., Tessone, C.J., Zenou, Y.: Nestedness in networks: A theoretical model and some applications. *Theoretical Economics* **9**(3), 695–752 (2014). DOI 10.3982/TE1348
- [10] Mahadev, N., Peled, U.: *Threshold Graphs and Related Topics*. North-Holland, Amsterdam (1995)
- [11] Rodríguez-Gironés, M.A., Santamaría, L.: A new algorithm to calculate the nestedness temperature of presence-absence matrices. *Journal of Biogeography* **33**(5), 924–935 (2006). DOI 10.1111/j.1365-2699.2006.01444.x
- [12] Saavedra, S., Stouffer, D.B., Uzzi, B., Bascompte, J.: Strong contributors to network persistence are the most vulnerable to extinction. *Nature* **478**(7368), 233–235 (2011). DOI 10.1038/nature10433

- [13] Soramäki, K., Bech, M.L., Arnold, J., Glass, R.J., Beyeler, W.E.: The topology of interbank payment flows. *Physica A: Statistical Mechanics and its Applications* **379**(1), 317–333 (2007). DOI 10.1016/j.physa.2006.11.093. URL <http://linkinghub.elsevier.com/retrieve/pii/S0378437106013124>
- [14] Tacchella, A., Cristelli, M., Caldarelli, G., Gabrielli, A., Pietronero, L.: A New Metrics for Countries' Fitness and Products' Complexity. *Scientific Reports* **2**, 1–4 (2012). DOI 10.1038/srep00723
- [15] Uzzi, B.: The Sources and Consequences of Embeddedness for the Economic Performance of Organizations: The network Effect (1996)

Clustering of Paths in Complex Networks

Mareike Bockholt and Katharina A. Zweig

Abstract While network analysis is more than 70 years old, the analysis of paths in complex networks is yet almost negligible. Here, we introduce different measures of computing the pairwise similarity of paths, either simply based on the elements in the paths, their sequence, on the graph in which they are embedded, or incorporating all three features. Based on ground-truth in a data set concerning how people solve a one-player puzzle, we show that the classification of the paths using the similarity measures in a hierarchical clustering approach performs best for the similarity measures which integrate all three features. We thus give first evidence that path similarity measures provide another dimension to mine and analyze complex networks.

1 Introduction

The analysis of complex networks has become a large and active field in which a broad variety of results has been published. In many cases, entities use the network as environment and move from node to node. The most obvious example is human navigation in spatial networks, travels in a transportation network, users surfing the WWW, but also game players exploring the problem space of the game, or students using an e-learning environment by following different paths through interlinked documents and media. In all these examples, the entities move on paths (or trails or walks) through the network which are usually neither the shortest path nor totally random (we will use the term *path*, if not explicitly stated otherwise, it includes walks and trails). But while there has been research concerned with human mobility patterns in a broad sense [4, 6], there has been almost no work which considers the actual *paths* taken. Consider for example the network shown in Figure 1 which shows which paths humans have taken in it. All humans navigating in this network started in the leftmost node and aimed at reaching the nodes in the bottom-right corner

Mareike Bockholt (e-mail: mareike.bockholt@cs.uni-kl.de) · Katharina A. Zweig (e-mail: zweig@cs.uni-kl.de)

Graph Theory and Complex Network Analysis Group, University of Kaiserslautern, Germany

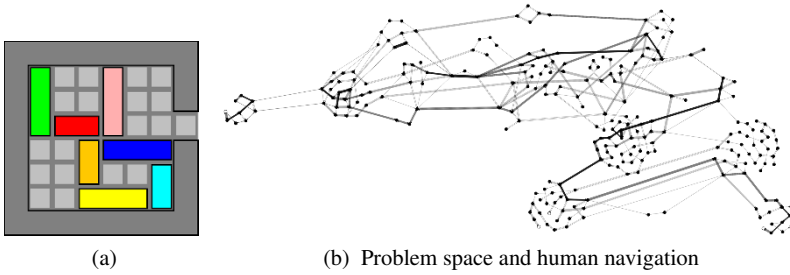


Fig. 1: (a) An example for a *Rush Hour* board. The red car needs to be removed from the board. A legal move consists of horizontal (vertical) move of one horizontally (vertically) placed car. (b) Each node represents one state of a puzzle and two states are connected by an edge if there is a legal move between them; some states represent the solution of the puzzle. The width of an edge is proportional to the number of users that made this move. Paths from a distinct starting state of the puzzle are called *solving* when they reach one of the states representing the solution of the puzzle.

of the picture. The thickness of the edges corresponds to the number of humans who used this edge in their path. It is astonishing that there are some paths in the network which are used more often than others although they are not necessarily the shortest ones. A human eye can also recognize that there are some paths which are more similar to each other than others. Also in other cases, it makes sense not to treat every path as a single path, but to find groups of similar paths and use these groups for further analysis. This can help to find common or distinguishing patterns in the paths and reduces the large amount of taken paths into representative groups. If such a clustering procedure is able to partition given paths into groups such that the paths within one group share elementary structural commonalities, it can be used in different application scenarios. By clustering paths of students in an e-learning environment, one might be able to identify different learner types and structure the materials accordingly. Grouping paths of players solving a puzzle can be used to find different strategies to solve the game. Clustering paths in a road network can lead to a procedure for identifying different means of transportation.

However, such a clustering requires a similarity measure. A similarity measure needs to be able to incorporate the most essential information contained in a path and weight them in an appropriate way. Therefore, the question arises of how to quantify the similarity of paths. It is surprising that there has been no approach proposed to measure the similarity of paths in complex networks and to group paths by similarity. Thus, in this paper, we: (i) provide seven first similarity measures for paths in networks which are either based on the elements contained in the paths, or on their sequence, on their embeddedness in the network, or on all three features, (ii) compute the proposed similarity measures for all pairs of paths of a benchmark data set with more than 13000 paths from 20 different networks (of the same kind), and (iii) for each of the networks, we cluster all paths with a hierarchical clustering approach with each of the proposed measures, and (iv) evaluate the results with

respect to a property of the paths that we set as ground-truth. It is crucial to note that this work does not aim at developing a classifier that partitions the paths according to the ground truth. This could be easily achieved by using other path-features or external features. The main goal is rather to evaluate the proposed similarity measures whether they are able to distinguish between structurally different paths.

The article is hence structured as follows: Section 2 gives an overview of research from other fields. Seven similarity measures for paths are introduced in Section 3. Section 4 gives the details of our approach for clustering paths, including the used data set (Sec. 4.1), the used ground truth and evaluation methods (Sec. 4.2), and the results (Sec. 4.3). Section 5 summarizes the findings of the article.

2 Related Work

While we know of no articles that proposed a similarity measure of paths in a complex network using their embeddedness in it, work that is related to the presented can be found in several different areas of research: In applications like video surveillance systems, it is desirable to track moving objects through consecutive video frames and to extract their trajectories. In order to automatically recognize anomalous movements of objects, a system needs to be able to distinguish between regular and anomalous trajectories. For this reason, there are several approaches how to compare and group trajectories of moving objects [1, 3, 15, 19]. The most often used similarity measures are the length of the longest common subsequence [3, 19] and the Hausdorff distance [12]. In the analysis of trajectories created from tracking moving individuals by (GPS) sensors, the Frchet distance has been extensively studied and applied [7], for example for detecting recurring patterns in trajectories [2]. In the context of web mining, it is beneficial to cluster similar user web sessions, for example for commercial or didactic interest, which is why there are several approaches to cluster sequential data. While Wang and Zaïane propose a clustering method for web sessions based on sequence alignment [20], Kumar proposes a new similarity metric for sequential data [13]. For comparing general sequential data, Moen, Mannila and Das presented several approaches [16, 17, 18] which use a measure similar to the longest common subsequence and eventually incorporates the similarity of the contained events themselves. Clustering of sequences has also been applied in order to make predictions, for example by Laasonen on routes of mobile phone users [14]. However, although some of these approaches can be adapted to paths, they do not consider the complex network in which the paths are embedded in. Taking into account the underlying complex networks is additional information which—as we will show in the following—will yield better results when finding groups of similar paths. Additionally, a systematic evaluation of possible similarity measures of paths has been not provided yet.

3 Similarity Measures for Paths

Definitions Let $G = (V, E)$ with $V = \{v_1, \dots, v_n\}$ and $E \subseteq V \times V$ denote a simple, connected, undirected, and unweighted graph. We define a path P in G as finite sequence $P = (p_1, e_{p_1}, p_2, \dots, p_{\ell-1}, e_{p_{\ell-1}}, p_\ell)$ with $p_i \in V$ for all $i \in \{1, \dots, \ell\}$ and $e_{p_i} = (p_i, p_{i+1}) \in E$ for all $i \in \{1, \dots, \ell-1\}$. Note that we do not require the edges or nodes of a paths to be distinct. Some authors would thus call P a walk. Since the considered graphs are simple, a path is uniquely determined by its node sequence and the notation can be simplified to $P = (p_1, p_2, \dots, p_\ell)$ which is used in the following. Let $V(P) = \{p_1, \dots, p_\ell\}$ and $E(P) = \{e_{p_1}, \dots, e_{p_{\ell-1}}\}$ denote the set of nodes and edges which are contained in a path P , respectively. The length $|P| = \ell - 1$ of a path P is defined as the number of (not necessarily distinct) edges. It holds that $|P| \geq |E(P)|$. Furthermore, let $I(P) = \{1, \dots, \ell - 1\}$ be the set of node indices of path P . For two nodes $v, w \in V$, we define the distance of v and w as the length of the shortest path between v and w . If there is no path from v to w , it is set $d(v, w) := \infty$. In the remainder of this article, we assume that G is a connected graph, hence $d(v, w) < \infty$ for all $v, w \in V$. For a path P and a node $v \in V$, we define the distance of v and P as $d(v, P) = \min \{d(v, w) \mid w \in V(P)\}$.

In the following, we assume that we have a graph G and a set of paths $\mathcal{P}(G)$ of valid paths in that graph. The research question is how to cluster these paths into coherent groups, given a suitable similarity measure $\sigma : \mathcal{P}(G) \times \mathcal{P}(G) \rightarrow \mathbb{R}$. In order to derive meaningful similarity and distance measures for paths, the most essential information contained in them needs to be determined. There are three obvious pieces of information contained in any path: (i) the elements contained in the paths, i.e., its nodes and edges, (ii) the order of the contained elements, and (iii) the position of the contained elements in the graph, i.e., their distance to the elements of the other path. Thus, as a first approach to determine the similarity of two paths, they can either be modeled as sets and existing measures for comparing sets can be used, or they can be modeled as sequences and existing measures for comparing strings or sequences can be used. Finally, paths can be considered as objects in the network, which allows incorporating the distance of the path's nodes in the graph into the similarity measure.

Element-based measures If a path is represented as a set of nodes or as a set of edges, well-known similarity measures for sets can be used, such as the number of common nodes or edges, or—as its normalized version—the Jaccard index [9]. The measures (*normalized*) *node set similarity* σ_{nss} (σ_{nss}^N) and (*normalized*) *edge set similarity* σ_{ess} (σ_{ess}^N) for two given paths $P, Q \in \mathcal{P}(G)$, are then defined accordingly (cf. Table 1).

Order-based measures If a path is understood as a sequence of nodes, similarity measures for sequences can be used, for example the *longest common subsequence* of the two paths [8]. For a path $P = (p_1, p_2 \dots p_{\ell-1} p_\ell)$, a subsequence of P is defined as any sequence of nodes which can be obtained by deleting nodes from P . Note that a subsequence of a path in a graph is not necessarily a valid path in that same graph anymore. For two paths P, Q , let $lcs(P, Q)$ denote the length of their longest common subsequence. The corresponding *LCS similarity* σ_{lcs} is as defined in Table 1, the

normalized similarity measure is obtained by dividing $lcs(P, Q)$ by the length of the longer path (see Table 1).

$\sigma_{nss}^{(N)}$	$ V(P) \cap V(Q) $	$\frac{ V(P) \cap V(Q) }{ V(P) \cup V(Q) }$	
$\sigma_{ess}^{(N)}$	$ E(P) \cap E(Q) $	$\frac{ E(P) \cap E(Q) }{ E(P) \cup E(Q) }$	
$\sigma_{lcs}^{(N)}$	$lcs(P, Q)$	$\frac{lcs(P, Q)}{\max\{ P , Q \} + 1}$	$lcs(P, Q)$ length of longest common subsequence of P, Q
$\delta_{sad}^{(N)}$	$\sum_{i \in I(P)} d(p_i, q_{G(i)})$	$\frac{\delta_{sad}(P, Q)}{\ell}$	G_{sad} identity function, $ P = Q = \ell - 1$
$\delta_{mad}^{(N)}$	$\begin{cases} \sum_{i=1}^{\ell} d(p_i, Q) & \text{if } \ell > k \\ \sum_{i=1}^k d(q_i, P) & \text{if } \ell < k \\ \min\{\sum_{i=1}^{\ell} d(p_i, Q), \sum_{i=1}^k d(q_i, P)\} \end{cases}$	$\frac{\delta_{mad}(P, Q)}{\max\{ P , Q \} + 1}$	$G_{mad}(i) = j$ s.t. $d(p_i, q_j)$ minimal, $ P = \ell - 1, Q = k - 1$
$\delta_{comappa1}^{(N)}$	$\min_{G \in \mathcal{S}_{comappa1}} \left\{ \sum_{i \in I(P)} d(p_i, q_{G(i)}) \right\}$	$\frac{\delta_{comappa1}(P, Q)}{\max\{ P , Q \} + 1}$	$ P \geq Q $, $\mathcal{S}_{comappa1}(P, Q)$ set of surjective and order-preserving functions $G : I(P) \rightarrow I(Q)$
$\delta_{comappa2}^{(N)}$	$\min_{G \in \mathcal{S}_{comappa2}(P, Q)} \left\{ \sum_{(i, j) \in G} d(p_i, q_j) \right\}$	$\frac{\delta_{comappa2}(P, Q)}{\max\{ P , Q \} + 1}$	$\mathcal{S}_{comappa2}(P, Q)$ set of left-total, right-total, order-preserving relations $G \subseteq I(P) \times I(Q)$

Table 1: Definitions of the similarity and distance measures for paths P, Q . σ and σ^N denote unnormalized and normalized measure in the first and second columns, respectively, similarly for distance measures δ .

Position-based measures While the previously proposed similarity measures only take into account nodes or edges contained in the paths or their order, we also propose four measures which consider the position of the paths in the network. The motivation is that even two paths that do not share a single edge can be close or distant within the graph they are embedded in. For example, if two people drive from the same city to the same other city, but one on a highway and one on country roads next to the highway, the two paths should be rated as more similar than if one drives from north to south and the other from east to west. The idea of the following measures is, thus, to calculate the distance in the graph from each node in P to a corresponding node in Q and to calculate the average of these node distances. A position-based distance measure for two paths P and Q is defined as $\delta(P, Q) = \sum_{i \in I(P)} d(p_i, q_{G(i)})$ for a mapping function $G : I(P) \rightarrow I(Q)$ which determines the counterpart for each node. The main problem is to find the appropriate counterpart of each node. A first naive proposal for G constrains the distance measure to paths with equal length and matches the i -th nodes of the paths with each other. For two paths P, Q with

$|P| = |Q| = \ell - 1$, G is set to $G_{sad}(i) = i$ for all $i \in \{1, \dots, \ell - 1\}$. This yields the (*normalized*) *simple average distance* as defined in Table 1. The simple average distance is a distance metric, but has two main deficiencies: it is only applicable to paths of equal length, and the matching function G might not be a good choice in many cases. For these reasons, we also consider the *matched average distance* which matches each node of P onto the node of Q which is closest by its graph theoretic distance. Since it seems reasonable to map each node of the longer path onto a node of the shorter path, we get for two paths P and Q with $|P| = \ell - 1$ and $|Q| = k - 1$ the measure δ_{mad} , as defined in Table 1. The normalized matched average distance δ_{mad}^N is obtained by dividing by the length of the longer path. For this distance measure, the corresponding mapping function is thus $G_{mad}(i) = j$ such that $d(p_i, q_j)$ is minimal. Note that with this mapping, it might happen that there are nodes in the shorter path which are not matched at all, although it is the shorter path of the two. Furthermore, while the simple average distance takes into account the order of the nodes in the path by the restrictive mapping G_{sad} , this quality is lost by weakening the restrictions to the node mapping. By mapping each node of P onto its *closest* node in Q (or vice versa), the mapping allows for example that the last node of P is mapped onto the first node of Q . It follows directly that this measure does not satisfy coincidence since two paths with identical node sets, but where the nodes occur in different order will have a matched average distance of 0 although they are not identical.

In order to avoid this, we require G to be a surjective function which considers the order of the nodes: we say that $G : I(P) \rightarrow I(Q)$ is *order-preserving* if for all $i, i' \in I(P)$, it holds that $i \leq i' \Leftrightarrow G(i) \leq G(i')$. Let $\mathcal{G}_{comappa1}(P, Q)$ be the set of all functions $G : I(P) \rightarrow I(Q)$ with these properties. The corresponding distance measure called (*normalized*) *CoMapPa1 distance* $\delta_{comappa1}$ (for COnsecutive MAPPING of PAths) is then obtained by taking the least expensive of these mappings (see Table 1). Note that $\mathcal{G}_{comappa1}(P, Q) = \emptyset$ if $|P| < |Q|$. A dynamic programming approach can be used to compute this measure in $\mathcal{O}((|P| - |Q| + 1) \cdot |Q|)$ assuming that the graph distances are precomputed.

The last distance measure to be introduced is a refinement of the CoMapPa1 distance leading to the CoMapPa2 distance measure. The CoMapPa1 distance measure exhibits an asymmetry because the longer path (P) is mapped onto the shorter path (Q): while each node of P is mapped onto exactly one node of Q , several nodes of P may be mapped onto one node of Q . In order to fix this issue, let $\mathcal{G}_{comappa2}$ be the set of all *relations* $G \subseteq I(P) \times I(Q)$ which are left-total, right-total, and order-preserving (where a relation G is *order-preserving*, if for all $(i, j), (i', j') \in G$, it holds that $i \leq i' \Leftrightarrow j \leq j'$). The corresponding distance measure, i.e., the (*normalized*) *CoMapPa2 distance* $\delta_{comappa2}$ ($\delta_{comappa2}^N$), is then defined as in Table 1. For two paths P and Q , this measure can be computed in $\mathcal{O}(|P| \cdot |Q|)$ using a dynamic programming approach, assuming the graph distances are precomputed.

Having these seven similarity and distance measures at hand, a data set of more than 13000 paths in 20 different networks is used to evaluate the proposed measures and give the proof of concept that clustering paths into groups is a viable way of mining complex networks.

4 Using the Measures for Clustering Paths

In Section 3, seven similarity (and distance) measures for paths are proposed (we will stick to the term *similarity measure*, if not explicitly stated otherwise, this term includes also the position-based measures although they are distance measures). The following approach clusters paths of a given data set by a hierarchical clustering approach, separately for each of the proposed similarity measures. We will give evidence that the similarity measure which incorporates information of the underlying complex network and the order of the nodes in the paths, i.e., the CoMapPa2 distance yield the most intuitive results for finding functional groups of paths. We start by providing information about the used data set before the method, the evaluation scheme, and the results are described.

4.1 Data

The networks of the data set are problem spaces of a board game such that the paths represent solutions of players. We consider the board game *Rush Hour* (invented by Nob Yoshigahara, distributed by ThinkFun Inc. and HCM Kinzel (Germany)) which is a one-player block sliding puzzle (see Figure 1a). It takes place on a board of 6×6 cells with one designated exit on which blocks are placed horizontally or vertically which represents a parking lot with parking cars. The blocks can have a length of 2 or 3 cells and a width of 1 cell. The goal of the game is to find a sequence of moves which allows a particular car to exit the board through the designated exit. A legal move is to move a car an arbitrary number of cells forwards or backwards, but not sideways. We call the exact positions of all cars a *configuration* of the game. We generate a graph $G^c = (V^c, E^c)$ from a *Rush Hour* start configuration c by taking all configurations reachable from the start configuration by legal moves as node set V^c , and the legal moves between them as edge set E^c . This graph is called the problem space associated to configuration c . We consider a *Rush Hour* game instance as solved when the cars on the board are in such positions that the particular car can be removed from the board with one additional move. We call such configurations *solution states*. With the concept of the problem space, solving a *Rush Hour* game instance can be understood as finding a path from c to a solution state. Such a path is called a solving path. In the optimal case, the found path is as short as possible.

Source The data set used for analysis was collected by Pelánek and Jarušek [11] who developed a *problem solving tutor* (available under tutor.fi.muni.cz) which is a web-based tool for learning by problem solving and is used in educational contexts. A detailed description is provided by Jarušek [10]. Among others, the system contains *Rush Hour* game instances of different degrees of difficulty. Twenty exemplary configurations with a sufficient amount of played paths were selected for analysis. Let \mathcal{C} denote this set of start configurations of the game instances. The data set contains the log data of all users of the system how they solved (or attempted to solve) the instances. It is important to note that users can also skip to the next game, if they feel they cannot solve the puzzle (or lose interest).

Preprocessing For each configuration $c \in \mathcal{C}$, the associated problem space G^c is computed¹. The problem spaces of the selected games are of the order of several thousands of nodes each. Any user who attempts to solve a game instance creates a path in the problem space of the configuration. For each user, each configuration and each attempt, the generated path is extracted from the log data. Any move which is done after a solution state was reached is not considered anymore, but the path is considered as solving path. Let \mathcal{P}_c denote the set of extracted paths for the configuration c . The table available under the given link also contains for each configuration how many paths were extracted (between 156 and 2934 paths) as well as the information of how many nodes of the problem spaces were actually visited by any of the players. Surprisingly, in average only 10% of the nodes were visited by at least one player.

Clustering For each of the configurations, for all pairs of paths from $\mathcal{P}_c \times \mathcal{P}_c$, all of the seven similarity measures are computed. For computing the simple average distance, the paths were cut to equal length for each configuration. However, in preceding studies for evaluating all similarity measures on the paths cut to equal length, the simple average distance has less promising results than the other distance measures. Thus, and because the simple average distance will be too restrictive for any application, the results for the simple average distance are omitted, and we only discuss the analysis of the complete uncut paths. The values of all unnormalized measures were scaled to the interval $[0, 1]$, the values of the similarity measures were then transformed by $1 - \sigma^{(N)}(P, Q)$ to result in a distance measure. For each configuration, the matrices with the similarity values for all pairs of paths are the input for an hierarchical clustering algorithm with either complete, average linkage methods or by Ward's clustering criterion [21]. The results for all three clustering methods show the same qualitative results and differ very little quantitatively; we thus only discuss the results of the clustering with complete linkage.

4.2 Ground Truth and Evaluation of the Results

For interpreting the results of the clustering procedures and to evaluate the different similarity and distance measures for paths, an evaluation criterion is necessary. For this, we use a very simple ground truth: a clustering procedure with an appropriate similarity measure as input should be able to distinguish between solving and non-solving paths. It is important to note that the goal of this work is not the development of a classifier which is able to distinguish between solving and non-solving paths. This could be done easily by other methods. The primary aim is to evaluate the presented similarity measures whether they are able to distinguish between structurally similar and dissimilar paths. In order to evaluate this, the semantic feature of the paths of being solving or non-solving is used: a well-

¹ A detailed description of the data set and the problem spaces can be found online under <http://gtma.cs.uni-kl.de/en/gruppe/bockholt/PDFs/CN2016SupplementaryMaterial.pdf>.

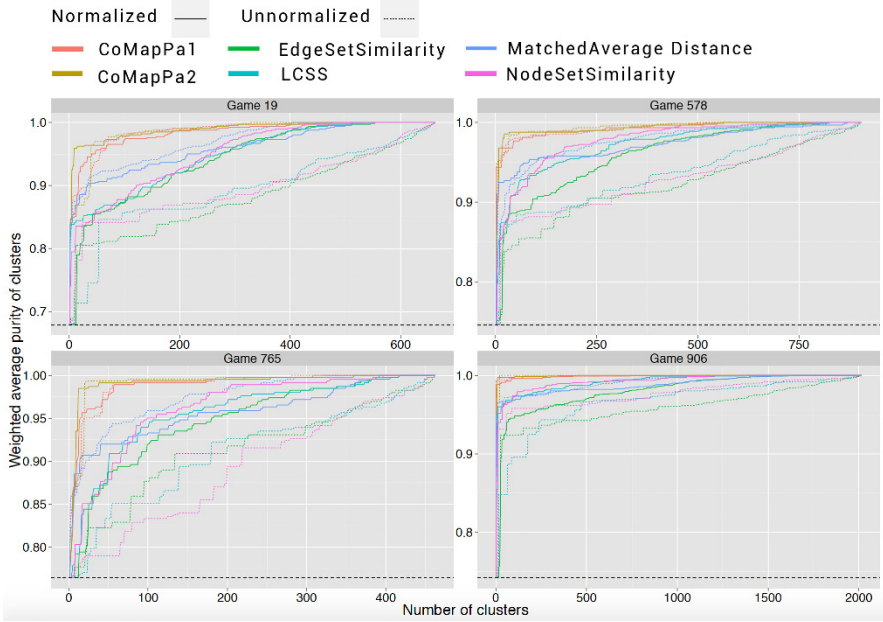


Fig. 2: Weighted average purity of the clustering results for some exemplary configurations, i.e., the Games 19, 578, 765, and 906.

designed similarity measure should at least distinguish between paths of these two classes. Hence, for each path of a configuration c , we define the binary attribute $q : \mathcal{P}_c \rightarrow \{0, 1\}$ which yields a 1 for a solving path, and a 0 for a non-solving path. A given cluster $\gamma = \{p_1, \dots, p_m\} \subseteq \mathcal{P}_c$ is then called *pure* if all paths in γ are either solving or non-solving. Since the requirement that a cluster should be pure, is a very strict one, we rather consider its *purity*. The purity of a cluster γ is defined as $\text{purity}(\gamma) = \frac{1}{|\gamma|} \max\{\sum_{p_i \in \gamma} q(p_i), |\gamma| - \sum_{p_i \in \gamma} q(p_i)\}$, i.e., the maximum of the two fractions of paths in γ which are solving or non-solving. Note that $\text{purity}(\gamma) \geq 0.5$ always holds. Let $q(\mathcal{P}_c) = \frac{1}{|\mathcal{P}_c|} \max\{\sum_{p \in \mathcal{P}_c} q(p), |\mathcal{P}_c| - \sum_{p \in \mathcal{P}_c} q(p)\}$ denote the fraction of paths for configuration c which are solving or non-solving.

For a given partition $\Gamma = \{\gamma_1, \dots, \gamma_k\}$ of \mathcal{P}_c , the average purity of all groups can be used as an evaluation criterion for the given partition. However, an unweighted average of the purities has the effect that the average purity is higher if Γ contains many singletons because they contribute with a purity of 1.0 each. We therefore consider a weighted average purity for Γ where the purity of each cluster from Γ contributes proportionally to its size to the average. The weighted average purity for a set of clusters Γ is defined as $\text{purity}_w(\Gamma) = \frac{1}{\sum_{\gamma_i \in \Gamma} |\gamma_i|} \sum_{\gamma_i \in \Gamma} |\gamma_i| \cdot \text{purity}(\gamma_i)$. However, the optimal number of clusters is not known. We thus consider the weighted average purity of all possible number of clusters. For a configuration c , the number of possible clusters ranges from 1 to $|\mathcal{P}_c|$. The weighted average purity for any configuration c

and for any similarity measure is 1.0 for $|\mathcal{P}_c|$ many clusters, and $q(\mathcal{P}_c)$ for 1 cluster. The behaviour between these extremes can then be used as evaluation criterion and means of comparison between the proposed similarity measures, for example to find out which similarity measure reaches the highest average purity with the smallest numbers of clusters.

4.3 Results

For each start configuration c and each similarity measure, the weighted average purity is computed for each number of clusters between 1 and $|\mathcal{P}_c|$. Figure 2 shows the results for some exemplary configurations. The possible number of clusters (i.e., the number of paths) is drawn on the x -axes, the corresponding weighted average purity of the clusters on the y -axes. Note that the weighted average purity is always larger than $q(\mathcal{P}_c)$ which is indicated by the dashed line. The first observation is that clustering with any of the similarity measures yields partitions with a weighted average purity considerably higher than the corresponding q value. Furthermore, the CoMapPa1 and CoMapPa2 distance measures perform clearly better than the purely set- or order-based measures. With these two measures, it is possible to obtain a weighted average purity close to 1 with only a few clusters. This observation is supported by Table 2 which presents the weighted average purity for the clustering results for all similarity measures for some graphs, if the number of clusters is fixed to 5, 10, 20, or 30^2 . For each game and for each $x \in \{5, 10, 20, 30\}$, the highest p_x is highlighted. Table 2 reveals that for almost all games, the CoMapPa1 and CoMapPa2 distance obtain the highest weighted average purity, often close to 100%. This is even achieved for game 723 where the number of solving and non-solving paths are almost equal. Nevertheless, clustering the 2704 paths with CoMapPa1 and CoMapPa2 yields almost pure clusters when only choosing 5 clusters. Figure 2 also indicates that the CoMapPa1 and CoMapPa2 measures perform almost equally well when using the normalized or unnormalized version of the measure. This is not the case for the set-based and order-based measures: here, the unnormalized measures consistently yield less good results.

In order to show that these observations are not only artifacts of single games, we adapt the idea of considering the area under the curve of the corresponding weighted average purity line. Informally, for a given sequence of weighted average purities (one entry per possible number of clusters) for one game and one similarity measure, we consider the area between the corresponding curve and the corresponding q line. Dividing this value by the size of the area of the “ideal” curve which reaches a weighted average purity of 100% with 2 clusters, yields the *relative AUC*. The relative AUC is computed for every similarity measure and every game. The results are shown in Figure 3 (left). The observations made for single games can be confirmed here. The relative AUC is consistently higher for all games for the CoMapPa1 and

² The table with the results for all configurations is contained in the supplementary material available under <http://gtna.cs.uni-kl.de/en/gruppe/bockholt/PDFs/CN2016SupplementaryMaterial.pdf>

Table 2: The weighted average purity for each of the six similarity measures for a fixed number of clusters. For each game, results for the unnormalized measure are presented in the first line, results for the normalized measure are presented in the second line. p_x denotes the weighted average purity of the clustering when choosing x clusters. For each game and each $x \in \{5, 10, 20, 30\}$ the highest p_x is highlighted. $q(\mathcal{P}_c)$ is denoted by q and gives the fraction of solving or non-solving paths of all paths for the configuration. All values are percentages. Because of lack of space, the table only shows the results for a few games. The full table is available online under the given link.

	σ_{nss}				σ_{ess}				σ_{ics}				δ_{mad}				$\delta_{comappa1}$				$\delta_{comappa2}$				q																								
	p_5	p_{10}	p_{20}	p_{30}	p_5	p_{10}	p_{20}	p_{30}	p_5	p_{10}	p_{20}	p_{30}	p_5	p_{10}	p_{20}	p_{30}	p_5	p_{10}	p_{20}	p_{30}	p_5	p_{10}	p_{20}	p_{30}		p_5	p_{10}	p_{20}	p_{30}																				
Game 19	69	69	78	84	69	74	81	81	68	71	71	71	87	87	88	89	85	88	89	90	85	85	87	88	79	79	84	84	68	68	81	84	84	84	84	85	84	86	89	89	85	85	92	94	92	96	96	96	67.82
Game 357	72	82	82	87	75	75	81	81	74	81	82	85	90	91	95	95	99	99	100	100	93	98	99	99	87	87	87	89	82	83	88	89	80	84	87	89	85	90	90	91	95	95	98	100	99	100	100	100	71.71
Game 723	55	56	66	74	55	57	58	63	55	57	65	79	95	95	96	96	99	99	99	99	99	99	99	99	74	90	94	94	55	56	58	61	81	84	93	94	95	95	96	96	96	99	99	99	99	99	99	99	54.44
Game 765	76	78	79	79	76	78	78	82	76	77	77	80	86	86	89	91	86	88	95	95	86	86	99	99	77	80	85	85	76	76	79	86	78	79	84	86	84	89	91	91	82	90	96	96	87	94	98	99	76.41

CoMapPa2 measure, regardless whether the normalized or unnormalized version is used. The relative AUC for all other measures is smaller and there are high differences between the normalized and unnormalized versions. When considering the results shown in Figures 2 and 3 (left), it is striking that the unnormalized versions of the set- and order-based measures yield clusters with a considerably smaller weighted average purity than the normalized version. There is the possibility that the similarity measures only distinguish between shorter and longer paths (because clearly, a solving path needs to have a certain length while non-solving paths can be short) and reach high average purity by this effect. Therefore, Figure 3 (right) shows the relative AUC of the resulting clusters, if for each game, only paths at least as long as the shortest solving path are considered. The gap between the normalized and unnormalized versions of the measures clearly decreases, but the general trend of the previous results is confirmed. Thus, clustering the paths with the proposed similarity measures can distinguish quite well between solving and non-solving paths. This implies that solving and non-solving paths show structural differences that can be detected by such simple similarity measures.

5 Conclusion

In this paper we have shown on a first benchmark data set and a simple ground truth, that already very simple quantifications of the similarity of paths in complex networks yield interesting insights into this new dimension of analyzable data. We have shown that—using a simple clustering algorithm—the measures which incorporate the underlying graph and the traversal order of the paths, contain the most information to categorize the paths representing the solving attempts of games into those that finally

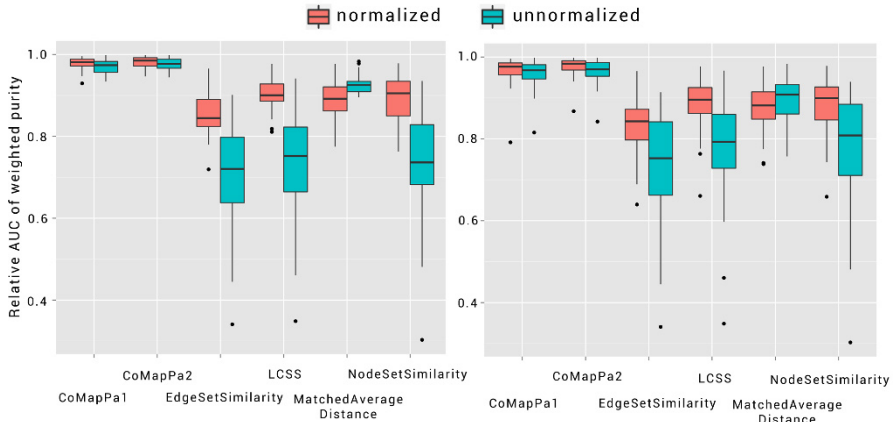


Fig. 3: Relative AUC of the weighted purity for all paths of all configurations (left) and when only sufficiently long paths are considered (right).

solve it and those that do not, to a quite high degree. The results imply that similarity measures which take into account the underlying network structure are best-suited to find groups of similar paths. However, the results are currently only valid for one specific data set which is why future work should aim at generalizing and validating the proposed measures on further data sets. In general, we believe that there is a wealth of data contained in the paths actually taken in a complex network rather than in the ones imposed by, e.g., centrality indices that always assume that either random walks or shortest paths are used. In another paper, Dorn, Lindenblatt and Zweig showed that centralities based on actual path data are also less prone to artifacts than classic centrality indices [5]. Thus, an important task for the community in network analysis should be to obtain such data and to publish it—preferably with ground truth regarding clusterings, centrality of nodes in the paths, external parameters like time taken or time stamps at the single nodes, etc.—to mine and analyze it together with the underlying network structures.

References

- [1] Bashir, F., Khokhar, A., Schonfeld, D.: Segmented trajectory based indexing and retrieval of video data. In: Proceedings of the International Conference on Image Processing, vol. 2, pp. II–623. IEEE (2003)
- [2] Buchin, K., Buchin, M., Gudmundsson, J., Löffler, M., Luo, J.: Detecting Commuting Patterns by Clustering Subtrajectories. In: Algorithms and Computation: 19th International Symposium, ISAAC 2008, Gold Coast, Australia, December 15–17, 2008. Proceedings, September, pp. 644–655 (2008)
- [3] Buzan, D., Sclaroff, S., Kollios, G.: Extraction and clustering of motion trajectories in video. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 2, pp. 521–524. IEEE (2004)
- [4] Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: user movement in location-based social networks. In: Proceedings of the 17th ACM SIGKDD international conference on

- Knowledge discovery and data mining, pp. 1082–1090. ACM (2011)
- [5] Dorn, I., Lindenblatt, A., Zweig, K.A.: The trilemma of network analysis. In: Proceedings of the 2012 IEEE/ACM international conference on Advances in Social Network Analysis and Mining, Istanbul (2012)
- [6] González, M.C., Hidalgo, C.A., Barabási, A.L.: Understanding individual human mobility patterns. *Nature* **453**(7196), 779–782 (2008)
- [7] Gudmundsson, J., Thom, A., Vahrenhold, J.: Of Motifs and Goals: Mining Trajectory Data. In: Proceedings of the 20th International Conference on Advances in Geographic Information Systems - SIGSPATIAL '12, pp. 129–138. ACM (2012)
- [8] Gusfield, D.: Algorithms on strings, trees and sequences: computer science and computational biology. Cambridge University Press, New York, NY, USA (1997)
- [9] Jaccard, P.: Etude comparative de la distribution florale dans une portion des alpes et du jura. *Bulletin del la Société Vaudoise des Sciences Naturelles* **37**, 547–579 (1901)
- [10] Jarušek, P.: Modeling problem solving times in tutoring systems. Ph.D. thesis, Masarykova univerzita, Fakulta informatiky (2013)
- [11] Jarušek, P., Pelánek, R.: Analysis of a simple model of problem solving times. In: S. Cerri, W. Clancey, G. Papadourakis, K. Panourgia (eds.) *Intelligent Tutoring Systems, Lecture Notes in Computer Science*, vol. 7315, pp. 379–388. Springer, Berlin Heidelberg (2012)
- [12] Junejo, I.N., Javed, O., Shah, M.: Multi feature path modeling for video surveillance. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 2, pp. 716–719. IEEE (2004)
- [13] Kumar, P., Raju, B.S., Krishna, P.R.: A new similarity metric for sequential data. *Exploring Advances in Interdisciplinary Data Mining and Analytics: New Trends: New Trends* p. 233 (2011)
- [14] Laasonen, K.: Clustering and prediction of mobile user routes from cellular data. In: Knowledge Discovery in Databases: PKDD 2005, *Lecture Notes in Computer Science*, vol. 3721, pp. 569–576. Springer, Berlin Heidelberg (2005)
- [15] Makris, D., Ellis, T.: Path detection in video surveillance. *Image and Vision Computing* **20**(12), 895–903 (2002)
- [16] Mannila, H., Moen, P.: Similarity between event types in sequences. In: Proceedings of the First International Conference on Data Warehousing and Knowledge Discovery, pp. 271–280. Springer, London (1999)
- [17] Mannila, H., Ronkainen, P.: Similarity of event sequences. In: Proceedings of the 4th International Workshop on Temporal Representation and Reasoning (TIME), p. 136. IEEE Computer Society (1997)
- [18] Moen, P.: Attribute, event sequence, and event type similarity notions for data mining. Ph.D. thesis, University of Helsinki, Department of Computer Science (2000)
- [19] Vlachos, M., Kollios, G., Gunopulos, D.: Discovering similar multidimensional trajectories. In: Proceedings of the 18th International Conference on Data Engineering, pp. 673–684. IEEE (2002)
- [20] Wang, W., Zaïane, O.R.: Clustering web sessions by sequence alignment. In: Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on, pp. 394–398. IEEE (2002)
- [21] Ward, J.H.: Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* **58**(301), 236–244 (1963)

Complexity Analysis of “Small-World Networks” and Spanning Tree Entropy

Raihana Mokhlissi, Dounia Lotfi, Joyati Debnath and Mohamed El Marraki

Abstract The number of spanning trees of a network is an important measure related to topological and dynamic properties of the network, such as its reliability, communication aspects, and so on. However, obtaining the number of spanning trees of networks and the study of their properties are computationally demanding, in particular for complex networks. In this paper, we introduce a family of small-world networks denoted $G_{k,n}$, characterized by dimension k , we present its topological construction and we examine its structural properties. Then, we propose the decomposition method to find the exact formula for the number of spanning trees of our small world network. This result allows the calculation of the spanning tree entropy which depends on the network structure, indicating that the entropy of low dimensional network is higher than that of high dimensional network.

Key words: number of spanning trees, complex network, small world network, decomposition method, spanning tree entropy.

1 Introduction

In nature, networks are everywhere around us. Owing to their relevance to many real systems, some of them are called complex networks. In recent years, they have been studied mainly focusing on fractals, scale free, small world [1, 13]... It could be applied to some real-world networks such as the world-wide web [6], social networks [9], mathematics, physics, etc... These networks contain a hierarchical property: “self-similarity” [11] which replicate their structure and their dynamics.

Raihana Mokhlissi (e-mail: mokhlissiraihana@gmail.com)✉ · Dounia Lotfi (e-mail: lotfi@fsr.ac.ma) · Mohamed El Marraki (e-mail: marraki@fsr.ac.ma)
LRIT associated unit with CNRST (URAC No 29)- Faculty of sciences, Mohammed V University in Rabat, B.P 1014, Rabat, Morocco.

Joyati Debnath
Winona State University, Winona, MN 55987, USA, e-mail: jdebnath@winona.edu

To analyze these complex networks, we need theories to understand their inherent and emergent properties [8]. We need new formal models of these networks so that we can predict accurately their performance, assert the guarantees of reliability, and ensures the survivability and the accessibility of communication. The graph theory has a powerful combinatorial tool to understand the relationship between the structure and the function of networks. This tool can be represented by a **Spanning tree** [14] which is one of the most important varieties of sub-networks to characterize the complex network constructions and understand their dynamical processes. It provides useful insights about the analyzing of the mechanism of self-similarity in complex networks. The notion of spanning tree is defined as a subgraph without cycle in other words a tree that has the same vertex as the main graph and some or all its edges. The applications of spanning trees of a network are often in computer networking. For example, if we have a redundant topology, the presence of loops generates broadcast storms that paralyze the network. To avoid routing loops, the spanning trees disable redundant links and restore the connection between the network nodes. In this work, Our goal is to determine the number of spanning trees of a network or what is called the complexity of a network [12]. The benefit of calculating this number is to evaluate the complexity of a network and to analyze its reliability [5]. This number can be obtained by computing the determinant of a submatrix of the Laplacian matrix corresponding to the network (Kirchhof's matrix-tree theorem [3]). However, for a large and complex network, the evaluation of this determinant is very difficult and even impossible. Most of the recent works have tried to find some alternative methods in order to avoid the tedious calculations of the largest determinant as needed by the algebraic method and enumerate the spanning trees for large and complex networks.

In this paper, we rely on the principle of a process of "Divide and Conquer" which divides a problem recursively in sub-problems, solves each of this sub-problems and then merges the partial results for a general solution. An example of this technique is **the decomposition method**: to calculate the number of spanning trees of a wide planar network, first, we represent it as graph and we cut it in two, three, ..., n subgraphs. Then, we calculate the number of spanning trees of each of subgraphs. Finally, we collect the results to obtain the complexity of the main graph. The use of this technique is due to its ease to discover the spanning trees of a complex network. In order that this method is relevant, we must investigate how we reduce the main graph and we have several possibilities to do it. In this work, we study the case where subgraphs are connected by one vertex (cut following one vertex).

In this article, we introduce a class of small world networks denoted $G_{k,n}$ where k is its dimension and n is the current iteration. This type of small world networks (SWNs) is a new model structures, which arises in the complex systems. Much attention has been paid to the study of this kind of SWNs, especially for the dimension $k = 3$, because it plays a notable role in the analysis of real-life complex systems [13], including the Internet, social networks, protein networks in the cell, tensor networks [10]... First, we present the construction of two models of SWNs: A particular case of the Small World Network $G_{3,n}$ having the dimension 3 and a general

case of the Small World Network $G_{k,n}$ having the dimension k . Then, we analyze their structural properties and we evaluate their complexity. Finally, we compute the entropy of their spanning trees which depends on their structure indicating that the entropy of low dimensional network is higher than that of high dimensional network.

2 Related work

The enumeration of spanning trees of a planar graph is not always easy, especially for a large graph. In order to facilitate this calculation, we propose a combinatorial technique which is based on the decomposition of graphs. This method aims to cut a graph in different parts or subgraphs satisfying certain constraints and optimizing a certain objective function. This partitioning problem has many applications such as clustering of documents, design electronic integrated circuits, load balancing for parallel machines and image segmentation. In this section, we define the decomposition method of a graph and its various combinatorial properties and we quote the main theorems which we needed to calculate the number of spanning trees for our network.

Definition 2.1. Let $G = C_1 \bullet C_2$ be a planar graph obtained by connecting C_1 and C_2 with one vertex v_1 . i.e., C_1 and C_2 are connected subgraphs which intersect exactly in one vertex v_1 (see Figure 1).

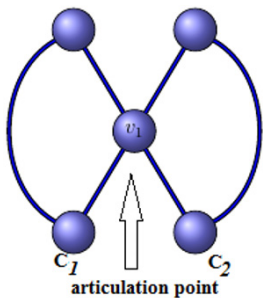


Fig. 1: A graph $G = C_1 \bullet C_2$

Property 2.1. Let G be a planar graph of type $G = C_1 \bullet C_2$:

- C_1 and C_2 have a common vertex v_1 and a common face (the external face).
- $V_G = V_{C_1} + V_{C_2} - 1$, $E_G = E_{C_1} + E_{C_2}$ and $F_G = F_{C_1} + F_{C_2} - 1$.
- If we remove the vertex v_1 of the graph G , the resulting graph is not connected.

Theorem 2.1. If we have a planar graph G such that $G = C_1 \bullet C_2$. Then, the number of spanning trees of G is given by:

$$\tau(G) = \tau(C_1 \bullet C_2) = \tau(C_1) \times \tau(C_2). \tag{1}$$

Proof. Each path that connects a vertex of C_1 to a vertex of C_2 must pass through v_1 . The Laplacian matrix associated with a graph $G = C_1 \bullet C_2$ is as follows:

$$L(\mathcal{C}) = \begin{pmatrix} v_1 & v_2 & v_3 & \dots & v_i & \dots & \dots & v_n \\ \begin{matrix} * & * & * & * & * & 0 & 0 & 0 \\ * & * & * & * & * & 0 & 0 & 0 \\ * & * & * & * & * & 0 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ * & * & * & * & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & 0 & * & * & * & * \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & * & * & * & * \end{matrix} & \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_i \\ v_n \end{matrix} \end{pmatrix}$$

After deleting the row and the column of the vertex v_1 , we obtain this matrix:

$$\begin{pmatrix} M_{n1,n1} & 0 \\ 0 & M_{n2,n2} \end{pmatrix}$$

In calculating the determinant, we obtain: $\tau(G) = \tau(C_1) \times \tau(C_2)$.

Theorem 2.2. (Generalization of Theorem 2.1) Let G be a chain of planar graphs defined by $G = C_1 \bullet C_2 \bullet \dots \bullet C_n$ (one of the following graphs in Figure 2). The number of spanning trees in G is given by the following formula:

$$\tau(G) = \prod_{i=1}^n \tau(C_i). \tag{2}$$



Fig. 2: Star graph and chain graph

3 The particular case of the Small World Network $G_{3,n}$ having the dimension 3

In this section, we introduce a most known kind of small world networks $G_{3,n}$ having the dimension 3. It has been extensively used quantum walks [2, 7], tensor networks [10]... $G_{3,n}$ is a particular case of a class of SWNs. We present its construction, determine their structural properties and analyze its complexity.

3.1 The construction and the structural properties of the Small World Network $G_{3,n}$

A class of small world networks denoted by $G_{3,n}$ with n is the current iteration is constructed as follows: At $n = 0$, we have a simple node. At first iteration, $G_{3,1}$ is a simple triangle. For $n > 1$, each node in the graph of the previous iteration is replaced by a new triangle. Thus, each of the newly appeared triangles contains exactly one node of the graph of the previous iteration. The growth process to the next iterations continues in a similar way. For illustration, in Figure 3, we present 4 iterations of $G_{3,n}$.

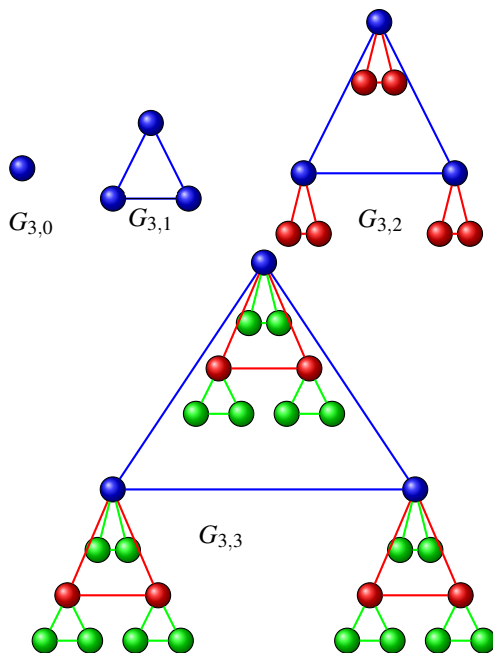


Fig. 3: A class of Small World Networks $G_{3,n}$ having the dimension 3

The structural properties of the small world network $G_{3,n}$ are presented as:

- The number of nodes of $G_{3,n}$ is calculated as follows: From Figure 3, we notice:
 $V_{G_{3,n}} = 3V_{G_{3,n-1}} = 3^2V_{G_{3,n-2}} = 3^3V_{G_{3,n-3}} = \dots = 3^{n-1}V_{G_{3,1}} = 3^nV_{G_{3,0}}$.
So the number of nodes of $G_{3,n}$ is: $V_{G_{3,n}} = 3^n$.

- The number of edges of $G_{3,n}$ is calculated as follows: From Figure 3, we notice:

$$\begin{aligned} E_{G_{3,n}} &= 3E_{G_{3,n-1}} + 3 \\ E_{G_{3,n-1}} &= 3E_{G_{3,n-2}} + 3 \\ E_{G_{3,n-2}} &= 3E_{G_{3,n-3}} + 3 \\ &\vdots \\ E_{G_{3,2}} &= 3E_{G_{3,1}} + 3 \\ E_{G_{3,1}} &= 3E_{G_{3,0}} + 3 \end{aligned}$$

We multiply the equation of $E_{G_{3,n-1}}$ by 3, the equation of $E_{G_{3,n-2}}$ by 3^2 and so on until the last equation $E_{G_{3,1}}$ which will be multiplied by 3^{n-1} . Summing all the obtained equations, we can find: $E_{G_{3,n}} = 3^nE_{3,0} + (3 \times 3^0 + 3 \times 3^1 + \dots + 3 \times 3^{n-1})$.
So the number of edges of $G_{3,n}$ is: $E_{G_{3,n}} = 3 \times \frac{3^n-1}{2}$.

- The number of faces of $G_{3,n}$ is calculated as follows: From Figure 3, we notice:

$$\begin{aligned} F_{G_{3,n}} &= 3F_{G_{3,n-1}} - 1 \\ F_{G_{3,n-1}} &= 3F_{G_{3,n-2}} - 1 \\ F_{G_{3,n-2}} &= 3F_{G_{3,n-3}} - 1 \\ &\vdots \\ E_{G_{3,2}} &= 3F_{G_{3,1}} - 1 \\ E_{G_{3,1}} &= 3F_{G_{3,0}} - 1 \end{aligned}$$

We multiply the equation of $F_{G_{3,n-1}}$ by 3, the equation of $F_{G_{3,n-2}}$ by 3^2 and so on until the last equation $F_{G_{3,1}}$ which will be multiplied by 3^{n-1} . Summing all the obtained equations, we can find: $F_{G_{3,n}} = 3^n - (3^0 + 3^1 + 3^2 + \dots + 3^{n-2} + 3^{n-1})$.
So the number of faces of $G_{3,n}$ is: $F_{G_{3,n}} = 3^n - \frac{3^n-1}{2}$.

3.2 Evaluation of the Complexity of the Small World Network $G_{3,n}$ having the dimension 3

The complexity of a complex network is very difficult to determine since classical approaches, such as the calculation of the determinant or the eigenvalues of the Laplacian matrix, are infeasible or even impossible for a large small world network. Therefore, we use the decomposition method that facilitate this computation to obtain the exact analytical expression for the number of spanning trees of the particular case of the small world network $G_{3,n}$.

Theorem 3.1. : *Let $G_{3,n}$ denote a class of small world networks having the dimension 3. The complexity of $G_{3,n}$ is given by the following formula:*

$$\tau(G_{3,n}) = 3^{\frac{3^n-1}{2}} \tag{3}$$

Proof. From the Figure 3, we see that $G_{3,n}$ contains several subgraphs as triangles G_3 . Using Theorem 2.2, we obtain: $\tau(G_{3,n}) = \prod^{T_{3,n}} \tau(G_3) = \tau(G_3)^{T_{3,n}}$ with $T_{3,n}$ is the number of triangles in $G_{3,n}$. From our network, we see:

$$\begin{aligned} T_{3,n} &= 3 \times T_{3,n-1} + 1 \\ T_{3,n-1} &= 3 \times T_{3,n-2} + 1 \\ T_{3,n-2} &= 3 \times T_{3,n-3} + 1 \\ &\vdots \\ T_{3,2} &= 3 \times T_{3,1} + 1 \\ T_{3,1} &= 3 \times T_{3,0} + 1 \end{aligned}$$

We multiply the equation of $T_{3,n-1}$ by 3, the equation of $T_{3,n-2}$ by 3^2 and so on until the last equation $T_{3,1}$ which will be multiplied by 3^{n-1} . Summing all the obtained equations, we can find: $T_{3,n} = 3^0 + 3^1 + 3^2 + \dots + 3^{n-2} + 3^{n-1}$. So the number of triangles in $G_{3,n}$ is: $T_{3,n} = \frac{3^n-1}{2}$. We replace it in the equation of $\tau(G_{3,n})$, hence we obtain: $\tau(G_{3,n}) = 3^{\frac{3^n-1}{2}}$. \square

4 The general case of the Small World Network $G_{k,n}$ having the dimension k

In this section, we study the general case of a class of small world networks $G_{k,n}$ having the dimension k . We examine its construction, analyze its topological properties and evaluate its complexity.

4.1 The construction and the structural properties of the Small World Network $G_{k,n}$

A family of small world networks denoted by $G_{k,n}$ is characterized by two parameters k and n , where k stands for the dimension of the cyclic graph and n for the current generation. The construction of $G_{k,n}$ is presented as follows: At $n = 0$, we have a simple node. At first iteration, $G_{k,1}$ is a simple cyclic graph with k nodes. For $n > 1$, each node in the graph of the previous iteration is replaced by a new cyclic graph with k nodes. Thus, each of the newly appeared cyclic graphs contains exactly one node of the graph of the previous iteration. The growth process to the next iterations continues in a similar way: Connecting a cyclic graph with k nodes to each node of the graph in the previous generation one gets the graph of the next generation. In Figure 4, we illustrate 4 iterations of $G_{k,n}$ with $k = 5$.

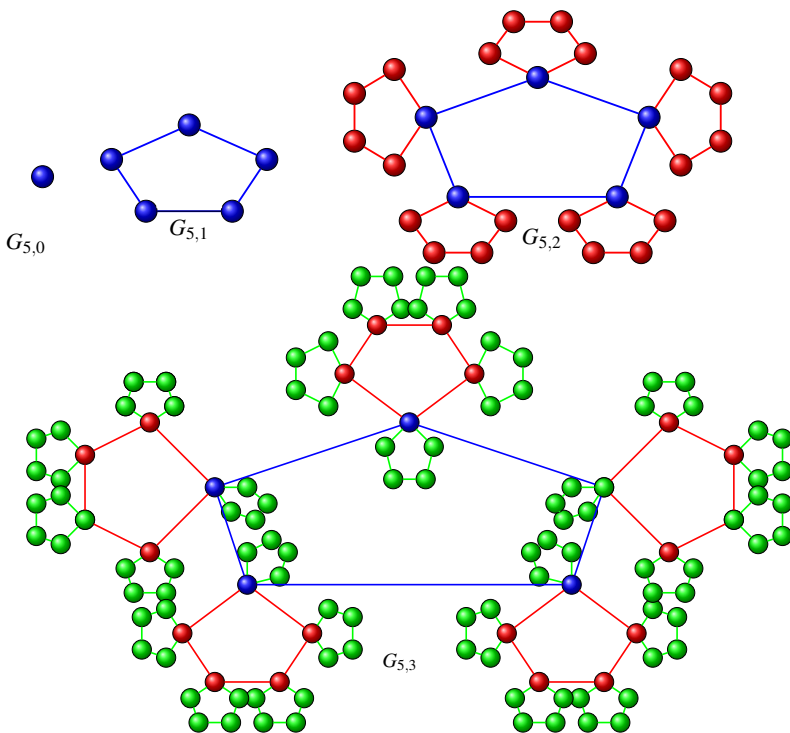


Fig. 4: A class of Small World Networks $G_{5,n}$ having the dimension $k = 5$

The structural properties of the small world network $G_{k,n}$ are presented as:

- The number of nodes of $G_{k,n}$ is calculated as follows: From Figure 4, we notice: $V_{G_{k,n}} = kV_{G_{k,n-1}} = k^2V_{G_{k,n-2}} = k^3V_{G_{k,n-3}} = \dots = k^{n-1}V_{G_{k,1}} = k^nV_{G_{k,0}}$.
So the number of vertices of $G_{k,n}$ is: $V_{G_{k,n}} = k^n$.
- The number of edges of $G_{k,n}$ is calculated as follows: From Figure 4, we notice:

$$\begin{aligned}
 E_{G_{k,n}} &= k \times E_{G_{k,n-1}} + k \\
 E_{G_{k,n-1}} &= k \times E_{G_{k,n-2}} + k \\
 E_{G_{k,n-2}} &= k \times E_{G_{k,n-3}} + k \\
 &\vdots \\
 E_{G_{k,2}} &= k \times E_{G_{k,1}} + k \\
 E_{G_{k,1}} &= k \times E_{G_{k,0}} + k
 \end{aligned}$$

We multiply the equation of $E_{G_{k,n-1}}$ by k , the equation of $E_{G_{k,n-2}}$ by k^2 and so on until the last equation $E_{G_{k,1}}$ which will be multiplied by k^{n-1} . Summing all the obtained equations, we can find: $E_{G_{k,n}} = k \times k^0 + k \times k^1 + \dots + k \times k^{n-1}$.

So **the number of edges of $G_{k,n}$ is:** $E_{G_{k,n}} = k \times \frac{k^n - 1}{k - 1}$.

- The number of faces of $G_{k,n}$ is calculated as follows: From Figure 4, we notice:

$$\begin{aligned}
 F_{G_{k,n}} &= k \times F_{G_{k,n-1}} - (k - 2) \\
 F_{G_{k,n-1}} &= k F_{G_{k,n-2}} - (k - 2) \\
 F_{G_{k,n-2}} &= k F_{G_{k,n-3}} - (k - 2) \\
 &\vdots \\
 F_{G_{k,2}} &= k F_{G_{k,1}} - (k - 2) \\
 F_{G_{k,1}} &= k F_{G_{k,0}} - (k - 2)
 \end{aligned}$$

We multiply the equation of $F_{G_{k,n-1}}$ by k , the equation of $F_{G_{k,n-2}}$ by k^2 and so on until the last equation $F_{G_{k,1}}$ which will be multiplied by k^{n-1} . Summing all the obtained equations, we find: $F_{G_{k,n}} = k^n - (k - 2)[k^0 + k^1 + k^2 + \dots + k^{n-2} + k^{n-1}]$.

So **the number of faces of $G_{k,n}$ is:** $F_{G_{k,n}} = k^n - (k - 2) \frac{k^n - 1}{k - 1}$.

4.2 Evaluation of the Complexity of the Small World Network $G_{k,n}$

According to the structural topology of the small world network $G_{k,n}$, we can apply the decomposition method following one node to obtain its number of spanning trees.

Theorem 4.1. : Let $G_{k,n}$ be a class of small world networks having the dimension k . The complexity of $G_{k,n}$ is given by the following formula:

$$\tau(G_{k,n}) = k \frac{k^n - 1}{k - 1} \tag{4}$$

Proof. From the Figure 4, we see that $G_{k,n}$ contains several cyclic subgraphs G_k . Using Theorem 2.2, we obtain: $\tau(G_{k,n}) = \prod^{T_{k,n}} \tau(G_k) = \tau(G_k)^{T_{k,n}}$ with $T_{k,n}$ is the number of cyclic subgraphs in $G_{3,n}$. From the figure 4, we see:

$$\begin{aligned}
 T_{k,n} &= k \times T_{k,n-1} + 1 \\
 T_{k,n-1} &= k \times T_{k,n-2} + 1 \\
 T_{k,n-2} &= k \times T_{k,n-3} + 1 \\
 &\vdots \\
 T_{k,2} &= k \times T_{k,1} + 1 \\
 T_{k,1} &= k \times T_{k,0} + 1
 \end{aligned}$$

We multiply the equation of $T_{k,n-1}$ by k , the equation of $T_{k,n-2}$ by k^2 and so on until the last equation $T_{k,1}$ which will be multiplied by k^{n-1} . Summing all the obtained equations, we can find: $T_{k,n} = k^0 + k^1 + k^2 + \dots + k^{n-2} + k^{n-1}$. So the number of subgraphs in $G_{k,n}$ is: $T_{k,n} = \frac{k^n - 1}{k - 1}$. We replace it in the equation of $\tau(G_{k,n})$ and $\tau(G_k) = k$, hence we obtain: $\tau(G_{k,n}) = k^{\frac{k^n - 1}{k - 1}}$. \square

Note: The small world network $G_{k,n}$ has the same number of nodes and edges as the dual Sierpinski gaskets [4], but they don't have the same complexity. This is due to the repositioning of nodes and how they are connected.

5 The entropy of spanning trees of a class of Small World Networks.

The asymptotic complexity or the entropy of spanning trees of a network G is a quantitative measure that compares the number of spanning trees of networks having the same average degree of nodes [9]. When $\tau(G)$: the spanning trees number of G grows exponentially with its number of vertices as $V_G \rightarrow \infty$, there exist a constant:

$$\rho_G = \lim_{V_G \rightarrow \infty} \frac{\ln|\tau(G)|}{|V_G|} \tag{5}$$

Let $\rho_{G_{k,n}}$ be the entropy of spanning trees for $G_{k,n}$. This real number is an interesting quantity characterizing the network structure. With the same average degree of the nodes $\langle z \rangle$ for a network, the bigger the entropy value, the more the number of spanning trees compared with other networks having the same average degree. We calculate and we compare the entropy of spanning trees of our SWN with other networks having the same average degree in order to determine the most reliable network with the strongest heterogeneous topology.

Corollary 5.1. : *The entropy of spanning trees of $G_{k,n}$ is: $\rho_{G_{k,n}} = \frac{\ln(k)}{k-1}$*

Proof. From the equation 4 and 5, and $V_{G_{k,n}} = k^n$, we obtain:

$$\rho_{G_{k,n}} = \lim_{V_{G_{k,n}} \rightarrow \infty} \frac{\ln(k^{\frac{k^n - 1}{k - 1}})}{k^n} = \lim_{V_{G_{k,n}} \rightarrow \infty} \frac{k^n(1 - \frac{1}{k^n})}{k^n} \times \frac{\ln(k)}{k - 1}, \text{ hence, } \rho_{G_{k,n}} = \frac{\ln(k)}{k - 1}.$$

According to the found formula of $\rho_{G_{k,n}}$, we see that this entropy is the same as that of Flower network, even if they don't have the same complexity. This result

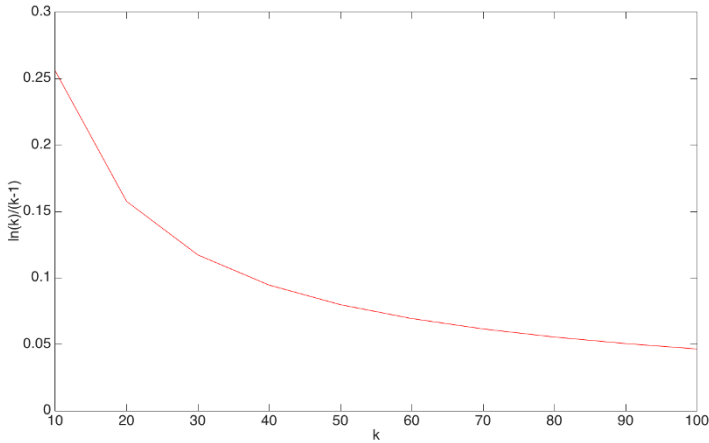


Fig. 5: The spanning trees entropy of $G_{k,n}$

shows that our model of SWN and the Flower network are similar in the limit $k \rightarrow \infty$ and they have similar behavior in this limit.

From Figure 5, we notice that the entropy of spanning trees of $G_{k,n}$ varies with the dimension k which shows that the spanning trees entropy depends on the basis of the self-similarity of our SWN (the network structure). Figure 5 also shows that the increasing of the dimension k leads to decrement the entropy of spanning trees of $G_{k,n}$. This indicates that the increase of the self-similarity dimension in our SWNs significantly decrease the number of spanning trees. To prove this result, we can compare the entropy of spanning trees of our SWN with different values of k with the entropy of other networks having the same average degree: The entropy of our SWN with $k = 2$ is (0,693) the highest reported for networks having the same average degree. The entropy of our SWN with $k = 3$ is (0,549) the same value that the entropy of the Hanoi networks [15]. The entropy of our SWN with $k = 5$ is (0.402) the lowest among all other networks having the same average degree 3, which means the entropy of low dimensional network is higher than that of high dimensional network. This reflects the fact that the low dimensional network of our SWN has more spanning trees than the high dimensional network. According all these results, we conclude that our class of small world networks $G_{k,n}$ having low value of dimension k is more robust and its structural topology has stronger heterogeneous than $G_{k,n}$ having high value of k .

6 Conclusion

Complex networks are an emerging and powerful tool that can be used in real-life complex systems. They are applied in communication networks, social networks, epidemiology, synchronization, etc... In this paper, we drew on ideas from graph theory to analyze structural properties and the complexity of a classe of small world

networks. We found its number of spanning trees by using the decomposition method. The knowledge of this number allows to calculate its spanning tree entropy indicating that the entropy of low dimensional network is higher than that of high dimensional network.

References

- [1] Amaral, L.A.N., Scala, A., Barthelemy, M., Stanley, H.E.: Classes of small-world networks. *Proceedings of the national academy of sciences* **97**(21), 11,149–11,152 (2000)
- [2] Anishchenko, A., Blumen, A., Mülken, O.: Enhancing the spreading of quantum walks on star graphs by additional bonds. *Quantum Information Processing* **11**(5), 1273–1286 (2012)
- [3] Chaiken, S., Kleitman, D.J.: Matrix tree theorems. *Journal of combinatorial theory, Series A* **24**(3), 377–381 (1978)
- [4] Chang, S.C., Chen, L.C., Yang, W.S.: Spanning trees on the sierpinski gasket. *Journal of Statistical Physics* **126**(3), 649–667 (2007)
- [5] Colbourn, C.J., Colbourn, C.: *The combinatorics of network reliability*, vol. 200. Oxford University Press New York (1987)
- [6] Cooper, C., Frieze, A.: A general model of web graphs. *Random Structures & Algorithms* **22**(3), 311–335 (2003)
- [7] Hillery, M., Reitzner, D., Bužek, V.: Searching via walking: How to find a marked clique of a complete graph using quantum walks. *Physical Review A* **81**(6), 062,324 (2010)
- [8] Jespersen, S., Sokolov, I., Blumen, A.: Relaxation properties of small-world networks. *Physical Review E* **62**(3), 4405 (2000)
- [9] Lyons, R.: Asymptotic enumeration of spanning trees. *Combinatorics, Probability and Computing* **14**(04), 491–522 (2005)
- [10] Marti, K.H., Bauer, B., Reiher, M., Troyer, M., Verstraete, F.: Complete-graph tensor network states: a new fermionic wave function ansatz for molecules. *New Journal of Physics* **12**(10), 103,008 (2010)
- [11] Song, C., Havlin, S., Makse, H.A.: Self-similarity of complex networks. *Nature* **433**(7024), 392–395 (2005)
- [12] Standish, R.K.: Complexity of networks (reprise). *Complexity* **17**(3), 50–61 (2012)
- [13] Watts, D.J., Strogatz, S.H.: Collective dynamics of small-world networks. *nature* **393**(6684), 440–442 (1998)
- [14] Wu, B.Y., Chao, K.M.: *Spanning trees and optimization problems*. CRC Press (2004)
- [15] Zhang, Z., Wu, S., Li, M., Comellas, F.: The number and degree distribution of spanning trees in the tower of hanoi graph. *Theoretical Computer Science* **609**, 443–455 (2016)

Graph Structure Similarity using Spectral Graph Theory

Brian Crawford, Raluca Gera, Jeffrey House, Thomas Knuth and Ryan Miller

Abstract In understanding an unknown network we search for metrics to determine how close an inferred network that is being analyzed, is to the truth. We develop a metric to test for similarity between an inferred network and the true network. Our method uses the eigenvalues of the adjacency matrix and of the *Laplacian* at each step of the network discovery to decide on the comparison to the ground truth. We consider synthetic networks and real terrorist networks for our analysis.

Keywords: graph comparison metrics, *Laplacian*, eigenvalue distribution, Kolmogorov-Smirnov Test.

1 Introduction and Motivation

The successful discovery of a network is of great interest to the Network Sciences community. Many algorithms have been proposed for network discovery. But when have we discovered enough of the Network? For a given network G , we utilize its subgraphs representing consecutive snapshots G_n ($1 \leq n \leq N$ with $G_N = G$), as G is discovered through monitor placement that light up the network. By lighting up G , we mean that certain nodes and edges of G are being discovered by using monitors on the nodes (monitors light up the node, its incident edges and adjacent vertices as defined in [6], while the remaining of the network is unknown as shown in Figure 1 for Boko and Noordin Top networks described in this paper. We compare consecutive snapshots (subgraphs) G_n at step n in the inference as the network is being inferred ($1 \leq n \leq N$). We present an analysis of the sequence of G_n to the

Brian Crawford

Department of Computer Sciences, Naval Postgraduate School, Monterey, CA

Raluca Gera (e-mail: rgera@nps.edu)✉ · Thomas Knuth · Ryan Miller

Department of Applied Mathematics, Naval Postgraduate School, Monterey, CA

Jeffrey House

Department of Operations Research, Naval Postgraduate School, Monterey, CA

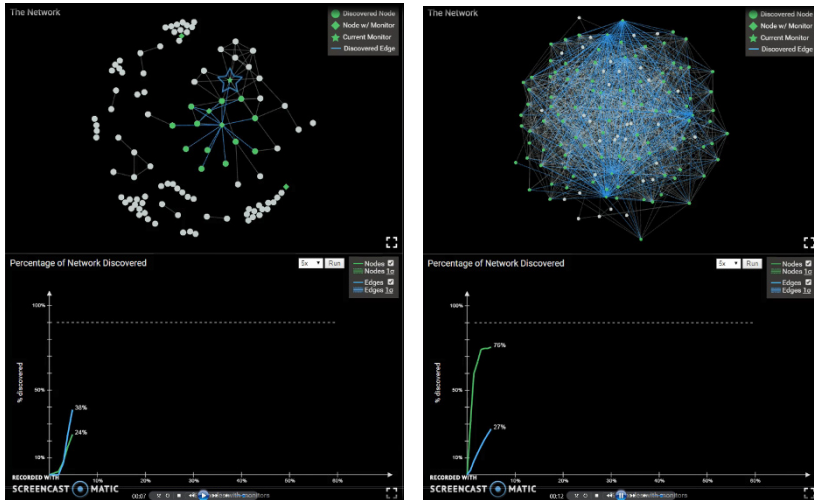


Fig. 1: Lighting up two dark networks: Boko and Noordin Top (click on the picture for the movie)

ground truth network $G = G_N$, which provides information about when enough of the network has been discovered. We develop a comparison metric using Sequential *Laplacian* and Adjacency Matrix Eigenvalue Distribution Comparisons. Four case studies, mixing synthetic and real terrorists networks, are explored in this paper to test the viability of the comparison metric. The first case study examines a synthetic network and the remaining case studies examine real terrorist (or dark networks) including Noordin Top [14], Boko Haram [4], and Fuerzas Armadas Revolucionarias de Colombia (FARC) [5].

2 Background

In mathematics, an established metric for graph comparison is isomorphism. Two labeled graphs G and H are isomorphic if there exists a one-to-one correspondence ϕ from $V(G)$ to $V(H)$ such that $uv \in E(G)$ if and only if $\phi(u)\phi(v) \in E(H)$ [2]. Comparing graphs based on isomorphism has a binary outcome: the graphs are either exactly the same (isomorphic), or they are different (non-isomorphic). However, in practice we prefer a measure that yields a range of similarity values for the non-isomorphic ones, and converges to 1 as we approach isomorphism.

Many methods were introduced to compare graphs: the original network reconstruction in systems theories started in the 1960s [10]. Intuitive approaches consider the percent of nodes and edges discovered during the inference of the network [6]. That is, they measure the percent of a network G that has been discovered at step n in network G_n through tracking $\frac{|V(G_n)|}{|V(G)|}$ and $\frac{|E(G_n)|}{|E(G)|}$. But these don't capture the

cardinality of **sets** of nodes and edges discovered, but not so much the **network**. The website <http://faculty.nps.edu/rgera/projects.html> [8] can be used to visualize the lighting up of the networks, and algorithms can be tested live on preloaded networks or custom networks, as desired by the user. The movie in Figure 1 was created using this website.

Other common metrics for measuring similarity use comparison of degree distributions, density, clustering coefficient, average path length, Maximum Common Subgraph, Graph Edit distance, number of spanning trees, and Hamming Distance. Many graphs have the same degree distribution, or clustering coefficient, and so on. Individually, none of these metrics comprehensively assesses topological similarity, rather each is some measure of node matching between networks. Methods that integrate all of these measures are desired. Similar efforts have been explored by mapping networks to vectors of the above properties, and then clustering the vectors based on naïve distance methods. However, the choice of features (and their weights if desired) is done manually which is not optimal.

Other methods to include Graph Kernels [9] which miss exactly the features presented above and more, such as community structures. Counting Graphlets [13] has been explored, but this is a computationally intensive technique. Other complementary techniques include Best-effort Pattern Matching [17], DeltaCon [11], Spectral analysis [19], and structural similarity of local neighborhoods [20]. A new research direction uses genetic algorithms and machine learning [1].

In this research we introduce two metrics to compare the topology of the networks using the eigenvalues of the Adjacency matrix and of the *Laplacian*. The question of interest in the network discovery problem is whether we have discovered the entire network. In general we cannot answer this question as it requires knowledge of "ground truth." However, it is always feasible to compare a sequence of discovered sub-graphs and analyze the similarity of neighbors in any sequence of sub-graphs. Spectral graph theory is concerned with understanding the structural properties of the graphs using the spectra or eigenvalues and eigenvectors of the graph. Eigenvalue analysis is used to describe the behavior of a dynamic system [18], and in our case, the behavior of a network representing the system. To see its relevance in comparing networks, note that eigenvalues measure the node cluster cohesiveness or community structure that has widely been studied in network science. Moreover, the eigenvalues represent the algebraic connectivity of the graph [7] and thus the spectra captures the topology of the graph. The largest eigenvalue and its corresponding eigenvector are of particular interest capturing the eigenvector centrality of nodes in a graph [3].

In spectral graph theory, Fan Chung [3] examined the distribution of eigenvalues of the graph. Most of this research is focused on the correlation of the range of the distribution of eigenvalues to the type of graph [3]. However, some research has been conducted on the behavior of the distribution of the eigenvalues of the graph. Mihail [12] suggests that there is a correlation between the power law distribution of the nodes of the graph and the distribution of the eigenvalues. In his analysis of several real graphs, including the Internet, he found that if the degrees of the graph $d_1 \dots d_n$ were power law distributed, then there is a high probability that the eigenvalues of the graph will be power law distributed and take on the values $\sqrt{d_1} \dots \sqrt{d_n}$ [12].

Of special interest for our analysis are eigenvalues the *Laplacian* $L = D - A$, where D is the degree matrix, and A is the adjacency matrix. Fan Chung supports the idea that the distribution of the eigenvalues of the *Laplacian* is more closely linked to the structure of the graph than only using the eigenvalues of the adjacency list [3]. The *Normalized Laplacian* (hereafter *Laplacian*) contains the degree distribution as well as the adjacency matrix information from the graph. While spectral analysis was previously used to cluster similar trees and synthetic graphs [19], we use the spectra with a different methodology.

Nonparametric statistical tests can capture whether two graphs are similar without actually knowing the true network. We compare two samples (subgraphs) and test the assumption they came from the same distribution (network). The alternative hypothesis is there is some type of change between the two samples, such as inferring more of a network. Ruth and Koyak introduce a new nonparametric test where the first m of N observations $X_1 \cdots X_m \cdots X_N$ are assumed to follow distribution F_1 and the rest are from F_2 . This allows us to see a “shift point” at X_{m+1} where our samples are no longer from the same distribution [15].

3 Methodology: Eigenvalue Distribution

One perspective on network discovery is to consider any subgraph as one of many possible outcomes from some discovery process, given a true underlying graph. For a simple graph $G(V, E)$, with $|V(G)| = n$, and $|E(G)| = \alpha$, there are 2^α possible subgraphs on N vertices. In real-world applications, say if $\alpha = 1200$, the count of possible subgraphs is grows rapidly: 2^{1200} is on the order of 10^{360} . Any discovered subgraph is one of many possible random outcomes. we search to determine how can we determine whether one collection of discovered nodes and edges is very similar to the underlying graph.

Let G_n be a sequence of graphs recorded while lighting up some given graph G , where, if $n < m$, then G_n was discovered before G_m , and $G_n \subseteq G_m, \forall n \leq m \leq N$. Let Λ_n be the list (or vector) of ordered eigenvalues for G_n , and let Λ be the vector of eigenvalues from the (true) underlying graph G . Note these are not eigenvectors - each is a vector of eigenvalues. Then if $G_N = G$, it follows $\Lambda_N = \Lambda$. During the process of discovering the network, we will not achieve $\Lambda_N = \Lambda$, but we expect that $\Lambda_n \rightarrow \Lambda$ as n increases. Similarly for the vector of eigenvalues of the Laplacian.

We conduct a numerical experiment to test whether we observe convergence of the *KS* test p -value in practice. We choose a graph, and using the Network Visualization Tool [8] we run a discovery algorithm as our method of establishing the sequence of nodes and edges discovered as shown in Figure 1. The algorithm is not relevant: it merely creates the sequence of subgraphs. We chose Fake Degree Discovery, a degree greedy algorithm that discovers the network using the degree of undiscovered nodes [16], see <https://github.com/Pelonza/Graph.Inference/blob/master/>. As discovery progresses, we obtain a sequence of graphs that get more similar to the ground truth, and can be used to validate our methodology.

We apply the Kolmogorov-Smirnov (KS) test, the nonparametric analog of the well-known chi-square test, to compare a sample of data to a known distribution and measure the “goodness of fit.” We assume the distribution of eigenvalues for each graph snapshot G_i arises randomly from a process driven by the structure of an underlying graph, rather than assuming observations are drawn from the same distribution.

We test the null hypothesis $\Lambda_n = \Lambda_m$ for $n < m$. For large steps values n and m , we expect that when the difference between n and m is small, that we would fail to reject this hypothesis. This leads to the conclusion that the subgraphs are similar. Note that failure to reject the null hypothesis does not imply the hypothesis is explicitly true. Rather, it means we have no evidence that it is false. Thus we should not conclude $\Lambda_n = \Lambda$ when we fail to reject the null hypothesis.

4 Results and Analysis

We discuss our experiments using a synthetic network in Section 4.1, and verify them by using our methodology on real terrorist networks in Subsection 4.2.

4.1 Numerical Experiment Outcomes on Synthetic Networks

We apply our algorithm to the base case graph: a randomly generated Erdős-Rény graph with 350 nodes and 3068 edges. When applying the Fake Degree Discovery

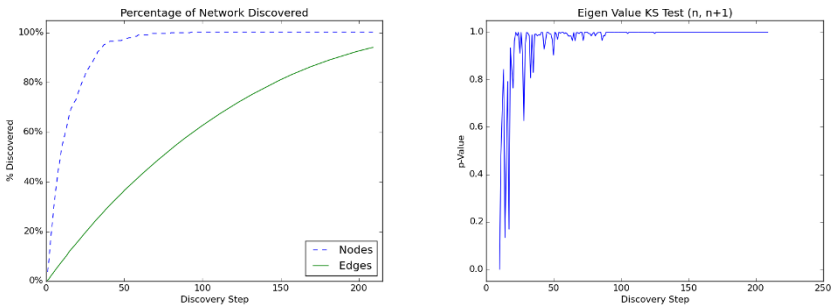


Fig. 2: Erdős-Rény: (a) Comparison of the Sequential Steps Plotting the Percent of Nodes/Edges (b) Comparison of the p -values of Sequential Steps (n against $n + 1$) using the Adjacency Eigenvalue Distribution

algorithm to this network, the maximum number of monitors placed to discover the whole network is 210, which is used as the “terminal” step for our plots. The first plot of Figure 2 shows the percent of nodes and edges discovered during this process.

In practice, network discovery is a sequential process and the true underlying graph is not available for comparison as done so far. Therefore we do not have the

luxury to compare against ground truth, and so we need to determine whether the KS test is useful when comparing sequential inferences. The second plot of Figure 2 shows that when only a few monitors are placed, many vertices and edges may be discovered in the graph, and thus p oscillates at first, being sensitive to the change in network from step n to step $n + 1$. While later in the discovery process, when a monitor discover little new information, the KS test has a high p -value, meaning the consecutively discovered graphs are very similar. The erratic behavior of the KS test p -values rapidly stabilizes through the inference, and remains high as expected.

In Figure 3 we plot the distribution of the adjacency matrix' eigenvalues for the graph obtained at step 20 alongside the graph at step 170, with each overlaid on the eigenvalue distribution at the terminal step. The x -axis is the index n of the eigenvalue λ_i of the adjacency matrix (notice that the eigenvalues are ranked in a non-increasing order, and the index is in an increasing order). Notice the difference in distributions between the two different time frames. Yet, the second plot shows almost identical distributions for a time frame closer to ground truth.

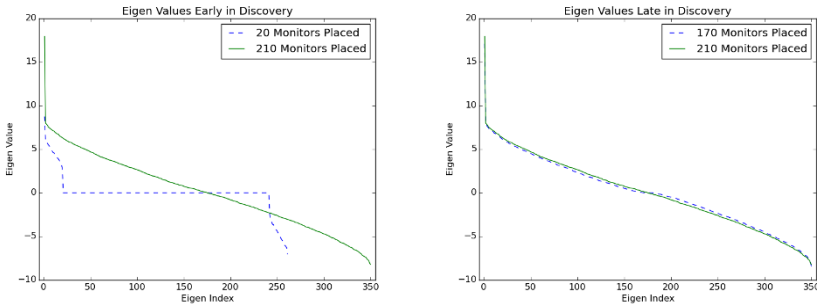


Fig. 3: Erdős-Rényi: Comparing the Adjacency Eigenvalue Distribution in the Discovery (at Step $t = 20$ and at Step $t = 170$) Against Ground Truth

The same comparisons for the Laplacian eigenvalue distributions for the same steps are shown in Figure 4. This dissimilarity is larger than the one obtained using the Adjacency matrix. This is due to a *Laplacian* matrix' capacity to capture more information about a graph's topology than an adjacency matrix. What is similar between the two graphs is the progressive convergence the early and late steps demonstrate. We will see that even for real networks, the late step is much closer than the early step to being aligned with the final graph.

Our final plots of the section shown in Figure 5 demonstrate the behavior of the KS test throughout the network discovery process against ground truth. In the first plot we see the adjacency matrix eigenvalue distribution is judged not to be similar until approximately step 150. Here we see a rapid climb from p -values near zero to p -values near one. The Laplacian eigenvalue distribution in the second plot shows very similar behavior, but the steep ascent of the p -values from zero to one occurs later, at step 190, as it is more sensitive to change due to the extra information captured by the Laplacian. At step 180 and step 190, 100% of nodes have been discovered. At

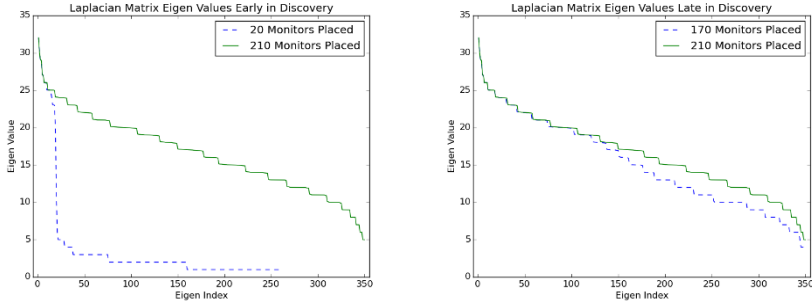


Fig. 4: Erdős-Rény: Comparing the *Laplacian* Eigenvalue Distribution Late in the Discovery (at Step $t = 170$) Against Ground Truth

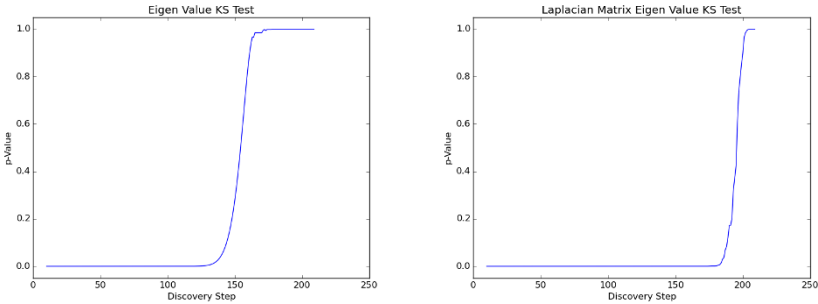


Fig. 5: Erdős-Rény: Adjacency Matrix and *Laplacian* Matrix Comparison of the Discovery Steps Against the Ground Truth

step 180, 88.8% of edges have been discovered, and at step 190, this rises to 90.8% of edges. Consider the impact of a missing edge when comparing the subgraph to the full underlying graph. In the adjacency matrix, a missing edge equates to two missing entries of value 1. But in the *Laplacian*, in addition to these missing entries, two diagonal entries representing the degrees of the nodes also differ from the full graph *Laplacian*. This explains why the KS test using the *Laplacian* is less likely to agree that the graphs are similar: there are additional sources of disagreement between the *Laplacians* not found in the adjacency matrices. We will compare the real networks to these plots, and analyze the similarities and dissimilarities.

4.2 Application to terrorist networks

We apply the methodology of Section 3 to three terrorist networks: Noordin Top [14], Boko Haram [4], and Fuerzas Armadas Revolucionarias de Colombia (FARC) [5].

4.2.1 Application to Noordin Top

Noordin Top Network (Figure 6) is the aggregation of 14 different relationship types amongst 139 terrorists for a total of 1499 edges. This network captures the relationships of five major terrorist organizations that operate in Indonesia. Noordin Top is the key broker between these organizations and exercises his influence to conduct large scale terrorist training events and operations. In this case, monitor placement during degree discovery process is representative of new information that is gained about the terrorist network. The plot of Figure 6 shows the node and edge progression of the discovery algorithm for a quick intuition of discovery.

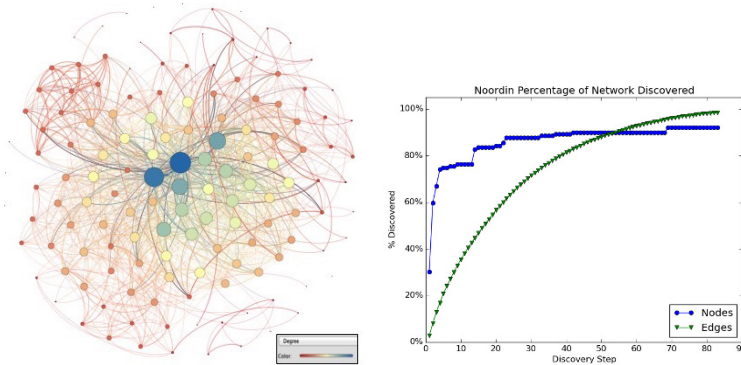


Fig. 6: Noordin Top and Its Inference: Comparison of the Percent of Nodes/Edges Discovery Steps Against the Ground Truth

Similar to Figure 5, we present the KS tests for the Noordin Top Network in the plot of Figure 7 and see the same behaviour. The second plot of Figure 7 also shows

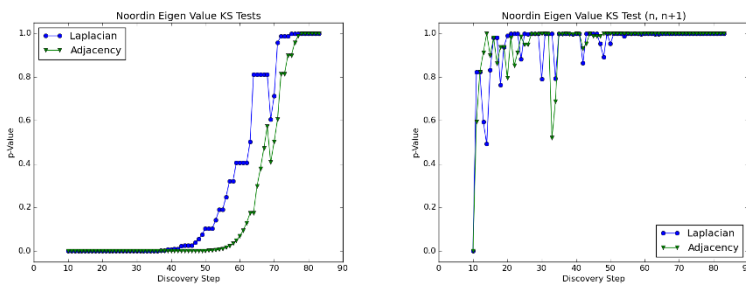


Fig. 7: Noordin Top Inference: (a) Comparison of the *Laplacian* and Adjacency Matrix Eigenvalues Steps Against the Ground Truth (b) Comparison of the *p*-values of Sequential Steps (*n* against *n + 1*) using the *Laplacian* and Adjacency Matrix

that consecutive graphs become more similar as the inference progresses, with more

noise than the synthetic network. The main differences are in the spikes seen in the both KS plots potentially due to the real network being disconnected.

4.2.2 Application to Boko Haram

The Boko Haram Network of Figure 8 and is the aggregation of 9 different relationship types (73 edges) amongst 105 terrorists. This network captures the relationships of an Islamic extremist organization that primarily operates in Nigeria. The plot in Figure 8 shows the node and edge progression of the discovery algorithm.

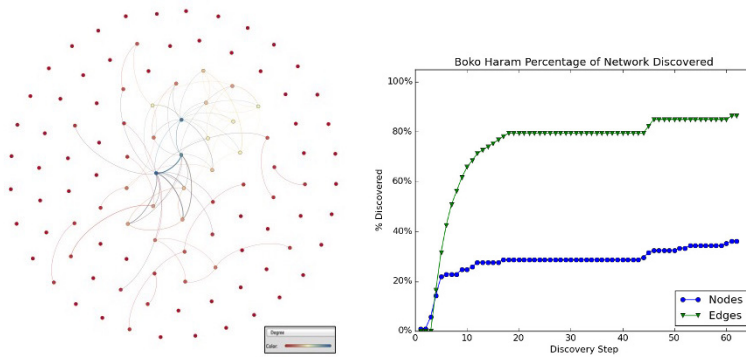


Fig. 8: Boko Haram and its Inference: Comparison of the Percent of Nodes/Edges Discovery Steps Against the Ground Truth

The KS tests plots for the Boko Haram Network against the ground truth and

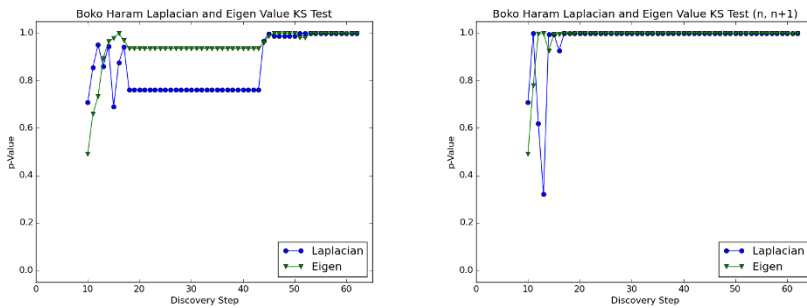


Fig. 9: Boko Haram Inference: (a) Comparison of the Eigenvalues Adjacency and Laplacian Matrix Steps Against the Ground Truth (b) Comparison of the p -values of Sequential Steps (n against $n + 1$) using the Adjacency and Laplacian Matrix

sequentially, for both adjacency and Laplacian matrices are shown Figure 9. Boko Haram is a disconnected network, with over 60 nodes of degree 0. When very few

nodes on this network are discovered, the p -value jumps very quickly. The drop in p -value at step 13 corresponds to a large discovery in the network that is less visible but detected in the edge and node discovery in Figure 8; and the p -value quickly stabilizes after.

4.2.3 Application to FARC

In applying our methodology to the FARC Terrorist Network for additional verification, we obtained similar results. The FARC Network is visualized as a network in Figure 10 and includes the aggregation of 10 different relationship types amongst 142 terrorists operating in Colombia, and a total of 1527 edges [5]. The plot in Figure 10 shows the node and edge progression of the discovery algorithm. We also plotted the KS tests for the FARC Terrorist Network in Figure 11. Here we note volatility in both the *Laplacian* and Adjacency KS test plots. This differs from the previous cases where we observed more stable convergence. We investigate this further to find an explanation.

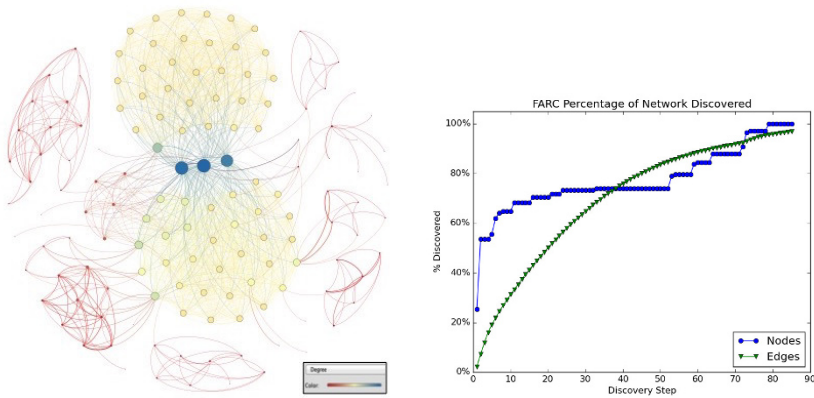


Fig. 10: FARC network and its Inference: Comparison of the Percent of Nodes/Edges Discovery Steps Against the Ground Truth

In the depiction of FARC in Figure 10 one observes two large, dense clusters, and several outlying clusters. We note that the FARC network is different from the other networks in what seems to be a crucial way: The clustering coefficient for this network is very high, at 0.91. The discovery algorithm focused on the big clusters at first (which can be seen in the plot of Figure 10 as the nodes get discovered quickly and then they plateau while only edges are being discovered), and then when nodes in a different cluster are discovered. The *KS* test detects and reports a “setback” in the confidence till the entire network has been discovered. Figure 11 shows that in the beginning the discovered graph is very dissimilar to the whole network as it has only a few edges discovered. The second plot of Figure 11 strengthens that explanation by showing that consecutive discoveries look more similar if the

eigenvalue of the Adjacency matrix is use, but the sensitivity of the *Laplacian* depicts the dissimilarities as it is more sensitive to changes in the graphs compared.

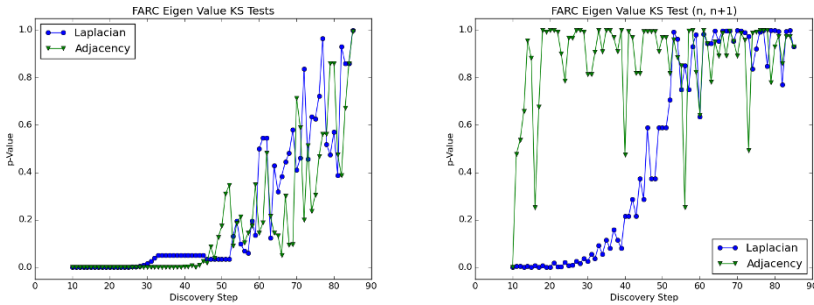


Fig. 11: FARC Inference: (a) Comparison of the *Laplacian* and Adjacency Matrix Eigenvalues Steps Against the Ground Truth (b) Comparison of the p -values of Sequential Steps (n against $n + 1$) using the *Laplacian* and Adjacency Matrix

4.3 Conclusions

Our numerical experiments show what we anticipated: Using the p -value from a KS test as a measure of similarity, the distribution of eigenvalues from neighboring sub-graph adjacency matrices are not always similar statistically, but this similarity measure stabilizes rapidly. Further, comparisons using this metric to the true underlying graph tend to 1 as the discovery unfolds.

When a representative portion of a graph has been discovered, the p value tends to stabilize. We base this statement on the rapid climb in the p -value for the KS test at some critical point, in each of the networks. Since the plots of the p -value, when comparing sequential steps of the inferred graph, show a steep climb in p -value at this critical point, which is the point to find a similar graph to ground truth.

We find this same very steep transition occurs much later for the *Laplacian*. There are also some known results on the distribution of eigenvalues from the *Laplacian*, including characterizations of graphs based solely on normalized eigenvalues. The *Laplacian* eigenvalue distribution comparison method is slower to conclude graphs are similar as it is armed with more information. We found that this delay is due to the structural differences in the adjacency matrix and the *Laplacian*. Thus for the purpose of similarity, the adjacency matrix can give a broad similarity measure, while the *Laplacian* is more exact in measuring similarity.

The rapid stabilization of the KS test when comparing consecutively discovered sub-graphs may offer some utility when comparing graphs in the setting where the true underlying graph remains unknown or unknowable. The advantage of such a metric is that it is self-referential: nothing needs to be assumed beyond what has been discovered. The desirable property of early stabilization can be put to use when

it fails: After the KS test measure on neighbors stabilizes, and discovery continues, a break in stability marks a major discovery. For example, if a bridge is discovered there is a clique on the other end of the bridge, then one can be sure the KS test p -value will drop. Whether it drops significantly will depend on the relative number of nodes and edges discovered in the next step compared to the number already discovered. We observed that when there is a high clustering coefficient, this leads to increased volatility in our similarity of measure.

We conclude that the use of sequential Adjacency and *Laplacian* matrix eigenvalue distribution comparisons based on the Kolmogorov-Smirnov Test p -values is a promising method to guide network discovery. Further work is necessary to explore and more fully describe the properties observed in this study. Particularly this method would not differentiate graphs that have the same graph spectrum (isospectral/cospectral graphs) as a theoretical study, as well as more choices of synthetic models.

Continuing the current research has great potential for comparing graphs and inferring networks when information is incomplete. A comparison to the Kullback-Leibler's (KL) divergence test can also be performed.

Acknowledgements The authors would like to thank the DoD for partially sponsoring the current research, and the reviewers for the valuable information they provided to us.

References

- [1] Aliakbary, S., Motallebi, S., Rashidian, S., Habibi, J., Movaghar, A.: Distance metric learning for complex networks: Towards size-independent comparison of network structures. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **25**(2), 023,111 (2015)
- [2] Chartrand, G., Zhang, P.: *A first course in graph theory*. Courier Corporation (2012)
- [3] Chung, F.R.: *Spectral graph theory*, vol. 92. American Mathematical Soc. (1997)
- [4] Cunningham, D.: The boko haram network. [machine-readable data file]. <https://sites.google.com/site/sfeverton18/research/appendix-1> (2014)
- [5] Cunningham, D., Everton, S., Wilson, G., Padilla, C., Zimmerman, D.: Brokers and key players in the internationalization of the farc. *Studies in Conflict & Terrorism* **36**(6), 477–502 (2013)
- [6] Davis, B., Gera, R., Lazzaro, G., Lim, B.Y., Rye, E.C.: The marginal benefit of monitor placement on networks. In: *Complex Networks VII*, pp. 93–104. Springer (2016)
- [7] Frankl, P., Rödl, V.: Forbidden intersections. *Transactions of the American Mathematical Society* **300**(1), 259–286 (1987)
- [8] Gera, R.: Network Discovery Visualization Project: Naval Postgraduate School network discovery visualization project. <http://faculty.nps.edu/dl/networkVisualization/> (2015)
- [9] Kashima, H., Inokuchi, A.: Kernels for graph classification. In: *ICDM Workshop on Active Mining*, vol. 2002. Citeseer (2002)
- [10] Klir, G., Elias, D.: *Architecture of systems problem solving*, 2nd edn., ifsr international series on systems science and engineering, vol. 21 (2003)
- [11] Koutra, D., Vogelstein, J.T., Faloutsos, C.: Deltacon: A principled massive-graph similarity function
- [12] Mihail, M., Papadimitriou, C.: On the eigenvalue power law. In: *Randomization and approximation techniques in computer science*, pp. 254–262. Springer (2002)

- [13] Pržulj, N.: Biological network comparison using graphlet degree distribution. *Bioinformatics* **23**(2), e177–e183 (2007)
- [14] Roberts, N., Everton., S.F.: Terrorist data: Noordin top terrorist network (subset). [machine-readable data file]. <https://sites.google.com/site/sfeverton18/research/appendix-1> (2011)
- [15] Ruth, D.M., Koyak, R.A.: Nonparametric tests for homogeneity based on non-bipartite matching. *Journal of the American Statistical Association* **106**(496) (2011)
- [16] Schmitt, K.: Fake degree discovery algorithm for lighting up networks. https://github.com/Pelonza/Graph_Inference/blob/master (2015)
- [17] Tong, H., Faloutsos, C., Gallagher, B., Eliassi-Rad, T.: Fast best-effort pattern matching in large attributed graphs. In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 737–746. ACM (2007)
- [18] Trefethen, L.N., Bau III, D.: *Numerical linear algebra*, vol. 50. Siam (1997)
- [19] Wilson, R.C., Zhu, P.: A study of graph spectra for comparing graphs and trees. *Pattern Recognition* **41**(9), 2833–2841 (2008)
- [20] Zager, L.A., Verghese, G.C.: Graph similarity scoring and matching. *Applied mathematics letters* **21**(1), 86–94 (2008)

A genetic algorithm-based approach to mapping the diversity of networks sharing a given degree distribution and global clustering

Peter Overbury, Istvan Z. Kiss and Luc Berthouze

Abstract The structure of a network plays a key role in the outcome of dynamical processes operating on it. Two prevalent network descriptors are the degree distribution and the global clustering. However, when generating networks with a prescribed degree distribution and global clustering, it has been shown that changes in structural properties other than that controlled for are induced and these changes have been found to alter the outcome of spreading processes on the network. This therefore begs the question of our understanding of the potential diversity of networks sharing a given degree distribution and global clustering. As the space of all possible networks is too large to be systematically explored, a heuristic approach is needed. In our genetic algorithm-based approach, networks are encoded by their subgraph counts from a chosen family of subgraphs. Coverage of the space of possible networks is then maximised by focusing the search through optimising the diversity of counts by the Map-Elite algorithm. We provide preliminary evidence of our approach's ability to sample from the space of possible networks more widely than some state of the art methods.

1 Introduction

Almost all complex systems can be modelled, to varying levels of detail, using networks whereby components of the system can be reduced down to nodes and to edges connecting them. Such an approach often makes it possible to pick out global behaviours dependent on the connections and/or relationships between different elements of the system that either would not have been noticed in isolation or could not be detected within large data sets [15]. The relationship between network

Peter Overbury (e-mail: po36@sussex.ac.uk) · Luc Berthouze (e-mail: L.Berthouze@sussex.ac.uk)✉

Department of Informatics, University of Sussex, Falmer, Brighton BN1 9QH,

Istvan Z Kiss (e-mail: i.z.kiss@sussex.ac.uk)

Department of Mathematics, University of Sussex, Falmer, Brighton BN1 9QH

structure and behaviour is the subject of much research in many areas such as epidemiology [3, 9, 18], social media [1] and neuroscience [12]. Where analytically-tractable mathematical models are needed, two main network descriptors stand out: degree distribution and global clustering. Interestingly, while there are now effective and analytically-tractable mathematical models that can handle the degree distribution well [3, 9, 18], when clustering is also considered, most models will break down or only operate for networks constructed in particular ways, e.g., networks with non-overlapping triangles [22]. This sensitivity to how networks are constructed highlights the fact that, as shown by [4, 8, 10, 19] among others, many network-generating algorithms introduce changes in structural properties other than that controlled for, thus undermining both model accuracy and inference of any causal role for the properties of interest. How to create network *null models*, i.e., where the properties of interest are fixed and all other properties are sampled in an unbiased manner, is an open question. One major step towards realising such goal would be to get a greater understanding of the space of networks satisfying a given set of requirements, e.g., a given degree distribution and a given global clustering coefficient. For networks of non-trivial size, the space of all such networks is too large to be systematically explored and therefore a heuristic approach is needed. Our approach relies on two principles: (a) a parametrisation of networks in terms of sub-graph decomposition, which significantly reduces the dimensionality of the encoding space when compared to the adjacency matrix as done in our previous work [17]; and (b) a search of the space driven by a process seeking to maximise the diversity of the networks being uncovered, thus biasing the exploration/exploitation trade-off toward exploration. The design and implementation of these two principles will be detailed in the following section.

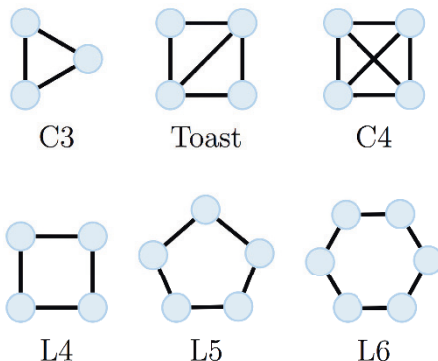
2 Methods

2.1 Network encoding

A key challenge in exploring the space of networks satisfying constraints is that of network representation. In principle, the network's adjacency matrix would be a natural choice because it fully specifies the network. However, it suffers from two major drawbacks: scalability and unicity (two networks may have a distinct adjacency matrix but be isomorphic). Our previous work [17] using the adjacency matrix revealed an extremely wasteful process even for small sized networks ($N = 200$). The recently-proposed dk-decomposition [16] offers an attractive alternative through its use of joint degree distributions of different orders, however, as we will show, questions remain regarding the biased nature of the network generation process once the joint degree distributions have been set. Instead, building on our recent work [20], we propose to parameterise networks in terms of a (arbitrarily chosen) family of subgraphs (see Figure 1 for a few examples).

Concretely, we use the counts of each of the subgraphs in the family to yield an adjacency matrix using the cardinality-matching algorithm (CMA hereafter) [20].

Fig. 1 The set of subgraphs used to encode networks (single edges not included). Subgraphs in the top row will induce clustering in the network.



CMA is a method inspired by the configuration model [6]. It assigns a set number of subgraphs of arbitrary structure in a network with a set degree sequence. Put simply, it works by assigning to nodes in the network hyperstubs of a certain degree as specified by each subgraph in the family. For example, triangles (subgraph C3) will require 3 hyperstubs of degree 2 whereas a Toast (see Figure 1 will involve 2 hyperstubs of degree 3 (corners with 3 edges) and 2 hyperstubs of degree 2 (corners with 2 edges). These hyperstubs are then selected at random and connected until there are no more left. When a new subgraph introduces self- or multi-edges, a new node is selected as in the matching algorithm [13]. When there is no option other than to add subgraphs over existing links or selecting multiple instances of the same node, the process is restarted from scratch. To accelerate the process, in this work, only 80% of the networks’ total edges were allocated to the specified subgraphs. The remaining edges were allocated as single edges to preserve the degree sequence. As this process can lead to nodes failing to have the desired degree (typically by ± 1), networks for which more than 20 nodes (out of a total of 1000) did not have the expected degree were excluded. Analysis of the networks produced (not reported here for reasons of space but available for an extended version, and see [20]) showed that the process still provides good control over most subgraphs, particularly (and advantageously in our context), those inducing clustering (i.e., C3, C4 and Toasts). Still, to avoid results being biased by a particular realisation, all measures reported in this paper were calculated by averaging over 5 network realisations. The reliability of the process is illustrated by Figure 2 which shows a compact spread of values of three network metrics (global clustering, mean shortest path length, mean betweenness centrality) for 10,000 realisations of a single network specification.

The choice of subgraphs is somewhat arbitrary and is a source of bias in itself. Here, we chose 3 subgraphs that induce clustering in the network (they are C3, C4 and toasts, see Figure 1). The other networks are loops that do not induce clustering. In this paper, only L4 and L5 were used. As a family, they provide flexibility and redundancy in the control for clustering. These 5 subgraphs have been shown in previous work to be those for which CMA showed most control over (as assessed by subgraph counting post realisation – results now shown here but available for an extended version).

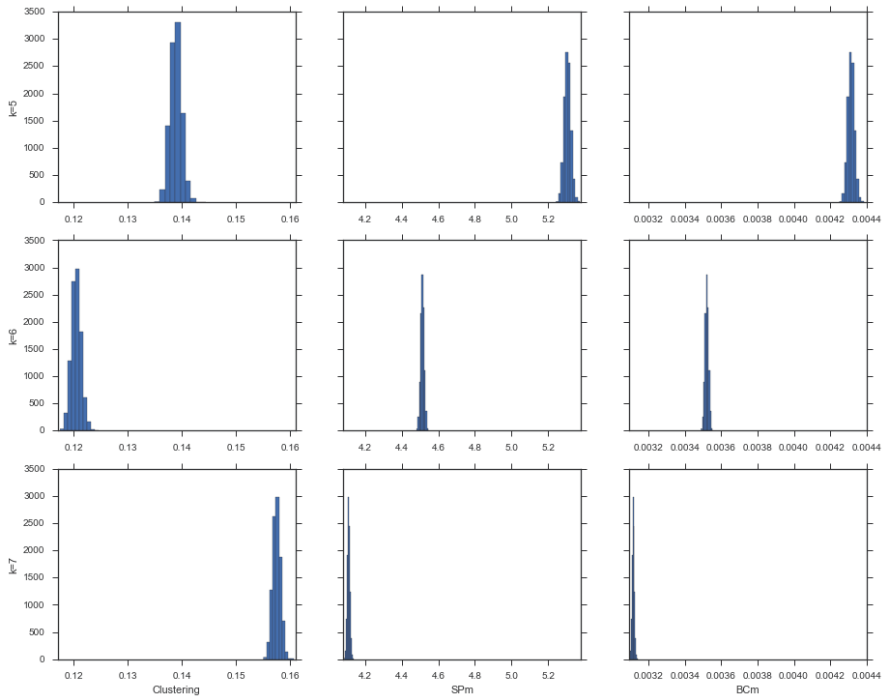


Fig. 2: Histograms of global clustering (left column), mean shortest path length (middle column) and mean betweenness centrality (right column) for 10,000 CMA realisations of a single network specification with predicted global clustering of 0.14 ± 0.025 . The top, middle and bottom rows correspond to regular networks with degree $k = 5$, $k = 6$ and $k = 7$ respectively.

2.2 Exploration of the space of possible solutions

Our primary objective being an exploration of the diversity of networks preserving a given degree distribution and global clustering coefficient, our task can be thought of as a two-part optimisation: (a) of the features that must be shared by a network for it to be added to the population of valid networks and (b) of the diversity within this population of valid networks. Multi-objective optimisation is not a new problem and the more complex variant considered here involving a changing measure of diversity within an actively changing population has recently been the focus of a number of methods in the field of genetic algorithms (GAs) [11].

In their simplest form GAs work by taking a starting population of individuals, which are encoded so that each has a *genome* that represents the key features being studied, here, the subgraph composition (expressed in percentage). This population is then *evolved* through *genetic operations* that change the genome of individuals. This typically involves *mutations* – the adding or subtracting from parts of the genome – and *recombination* or *crossover* – the combining of two individuals into a new individual with a new genome. Here, mutations involve changing the prevalence of

each subgraph by a small number drawn randomly in the interval $[-0.1, 0.1]$. During crossover between two networks, a new network is created whereby a randomly chosen number of its subgraph percentages are those of the first network and all others are those from the second network. For both mutation and crossover, the subgraph prevalences of the new individual are normalised to sum up to 1. Both processes have a 60% chance of occurring to either an individual (for crossover) or an individual subgraph count (for mutation) at each generation. All individuals are then analysed for their *fitness* – the objective function in the optimisation process, here, global clustering calculated using the formula proposed in [7]. Those with the lowest fitness are either removed, selected for genetic operations less often or both. This results in a population that, depending on the setting of the GA, moves along the search space towards areas of high fitness. An important implication is that the solutions are highly dependent on the choice of the fitness measure, the selective pressures used at each generation and the way that solutions are stored.

Previous work based on the idea of optimising for diversity includes the generation of neural networks topologies for control of robots in which diversity of both behaviour and performance was optimised for [21] and our own work [17] in which we started exploring the feasibility of using GAs to optimise the diversity of networks satisfying structural constraints, albeit for small sized networks. The main limitation of these methods has been their focus on the optimisation of a few individuals to the best possible fitness over all their objectives (the Pareto front), often leading them to avoid equally valid/fit regions of the feature space. Here, we employ the recently proposed Map-Elite method [14] which seeks to map the solution space through dividing the space into identically-sized multi-dimensional *cells* that cover a set range of values for each of the features used to describe the individuals. All individuals in the population are then placed in one of these cells and when new individuals are created they are assessed based only on individuals in that same part of the space. If there is no other in the cell then the individual is deemed novel and is kept. If, instead, there is another individual already within the cell then only the individual with the greatest fitness is kept. This method allows for the promotion of novelty without comparison of the entire population whilst also optimising the fitness of the population.

3 Results

The experiments reported in this paper sought to map the diversity of networks of size $N = 1000$ satisfying the constraint of a homogeneous/regular degree distribution (with degree 5, 6 or 7 – as three distinct scenarios) and a global clustering coefficient of 0.14. Although our choice of network encoding is insensitive to network size, the CMA connection process is not. The size $N = 1000$ makes the experiments tractable, when deployed on the high performance computing facility. The three degrees considered enable us to assess the effectiveness of the method for networks with more ($k = 7$) or less ($k = 5$) flexibility in how to allocate subgraphs. For example, with $k = 5$, it would not be possible for a node to share a fully connected

square (C5) and the degree 3 corner of a toast whereas with $k = 7$, the same node could accommodate that and an extra free edge. Our choice of global clustering coefficient is arbitrary although one should note that depending on the choice of subgraph family used to encode networks, some clustering values are more likely than others. With the proposed family of subgraphs and the relatively small degree, it would be difficult to generate highly clustered networks, and diversity would be extremely limited. A tolerance of ± 0.025 was used in evaluating the clustering fitness of networks. A tolerance is needed due to (a) the nature of the computation of the clustering coefficient and (b) the stochasticity in allocating subgraphs and any resulting byproducts [20]. This tolerance, which is reflected in the histograms of clustering values in Figure 2, corresponds to a maximum deviation of ± 8 triangles (subgraph C3) from the expected number of subgraphs and is negligible given the number of triangles needed to achieve the required clustering.

3.1 Effectiveness of the mapping in terms of space coverage

To provide some quantitative assessment of the effectiveness of mapping, cells were configured for maximal resolution, meaning that all individuals within a cell would have the exact same subgraph counts. It should be noted at the outset (but this is currently the subject of further work) that starting out with maximal resolution is sub-optimal in terms of managing the evolutionary process. However, for the purpose of this assessment, it provides as detailed a picture as possible of the proportion of all possible encodings that is uncovered by the evolutionary process (with the caveat that with a limited number of generations, the actual number of cells uncovered can only be a tiny fraction of the total number of cells possible). In the following, when ignoring the fact that not all combinations of subgraph counts are actually realisable – graphicality of the network), the total number of cells possible is $1040625000000 = 333 \times 250 \times 250 \times 200 \times 250$ and corresponds to the product of the ranges of possible values taken by the counts of each subgraph in the family (this count is determined on the basis of the highest-degree hyperstub in relation to the total number of nodes available in the network). The actual total number of cells is found by subtracting from the above count those cells that correspond to non-graphical/non-realizable networks, namely, those where the total number of edges prescribed by the subgraph decomposition is above $(Nk)/2$ and where the number of triple hyperstubs from C4 and Toasts is greater than $(k/3)N$ – the maximum number of triple hyper stubs allowed by CMA in a network. Coverage of the space at various points during the process is shown in Table 1. Given the maximum resolution and the fact that each generation only produces one new network, the actual percentage of coverage is very small. However, the table shows two important results: (a) the rate at which new cells are explored in relation to the number of generations is almost 1 suggesting that cells are not revisited (this would no longer be the case if cells had lower resolution); (b) the rate at which valid networks are produced is roughly constant as the number of generations increases.

k	21,000 gen		42,000 gen		63,000 gen	
	Explored	Valid	Explored	Valid	Explored	Valid
5	20783	12995	41546	25952	62286	38852
6	20824	18266	41583	36596	62349	55009
7	20845	18691	40646	36680	62431	56435

Table 1: Number of explored and valid cells uncovered by the evolutionary process at various time points for the three scenarios ($k = 5, 6, 7$) considered. In all cases, networks have size $N = 1000$ and the family of subgraph considered is (C3, C4, Toast, L4 and L5) with a desired global clustering of 0.14 ± 0.025 . For reference, the total number of cells possible (after removal of non-graphical solutions) is $\sim 10^{12}$. Each generation can produce at most one new network.

Importantly, we note that this table does not provide any information regarding coverage of the space of valid networks, those with correct degree distribution and global clustering within ± 0.025 of the desired clustering. Whilst the search is focused on finding valid cells (rather than all possible cells), we do not have any estimate for the total number of possible valid networks in the space of all possible networks. Figure 3 provides a different perspective on this by using low-dimensional projections of the space of networks explored and valid. Where possible, non-graphical solutions have been highlighted. The Figure reveals that despite the limited number of generations (again, corresponding to a very small percentage of all possible configurations) there is evidence of fairly uniform sampling as far as explored cells are concerned. The Figure further reveals pair-wise relationships between counts of subgraphs that reflect the constraints of the problem. For example, when two clustering-inducing subgraphs are considered (e.g., C4 and Toast) there is a distinct relationship whereby configurations with larger numbers of C4s have smaller numbers of Toast and conversely. Instead when clustering-inducing subgraphs and non clustering-inducing subgraphs are considered (e.g., C3 and L4) valid configurations can be found throughout the space of explored solutions. Areas that are not explored are typically reflecting configurations for which although no graphicality condition is being violated as far as the particular pair of subgraphs is concerned, no network realisation is possible when taking into account the other dimensions.

3.2 Comparison with other methods

Whilst the above results point to evidence of diversity in terms of subgraphs a more useful basis for evaluating the effectiveness of our approach is to assess the extent to which networks uncovered show greater diversity than can be expected from methods currently available to generate networks satisfying the same constraints. Since subgraphs counts are explicitly controlled by the evolutionary process, they would not be a fair metric for comparison. Instead, we considered two global structural properties:

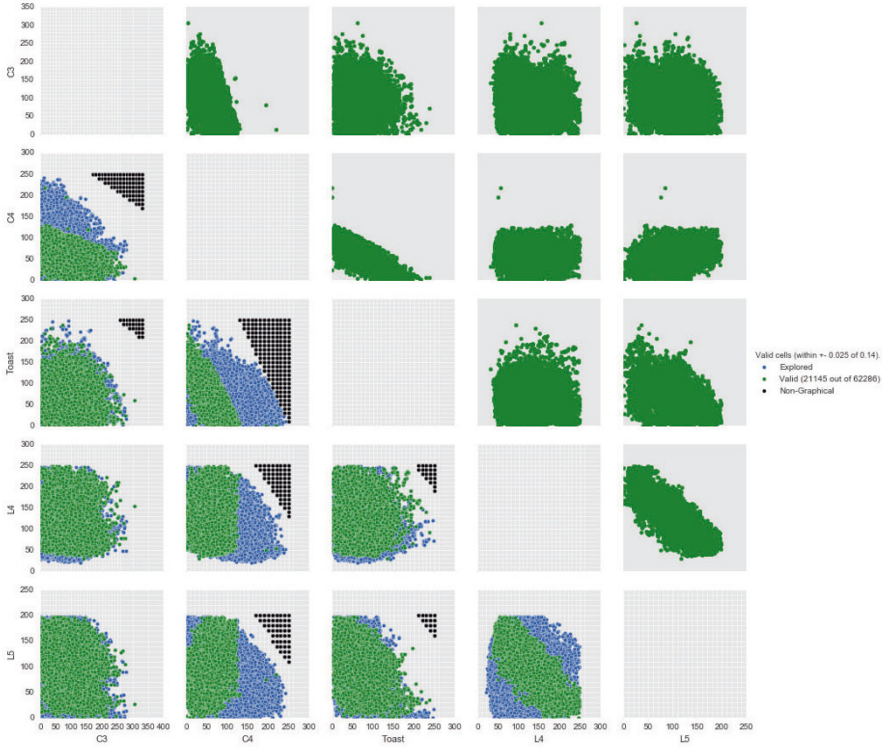


Fig. 3: Low-dimensional projections of the configurations discovered by the evolutionary process (both those that were explored but not necessarily satisfying the constraints – in blue – and those that were valid – in green) after 63040 generations. Each dot denotes a network whose coordinates are the counts for the subgraphs shown in the horizontal and vertical axes. A dot does not define a unique network, however, as the projection can mask great diversity in the remaining 3 dimensions.

mean shortest path length and mean betweenness centrality (BC_m) – although as both show a high degree of correlation, only betweenness centrality will be reported below. These properties are important determinants of behaviour in networks [15]. Two state of the art network generating methods have been used for this comparison: dk-series decomposition [16] and BigV rewiring [5]. For the former, we used dk2.1 (using code from [2]) which preserves degree distribution and global clustering (dk2.5 would also preserve local clustering which is overly specific for our purpose). Since the dk method requires a seed network to operate, one network was chosen at random among those generated by our approach. For the latter, the rewiring algorithm was applied to a single random network with homogeneous degree distribution who was rewired until desired clustering was achieved (with a maximum of 40000 rewirings). For both BigV rewiring and dk decomposition, the number of networks generated was set to the number of networks produced by the GA.

Figure 4 reveals that the range of mean betweenness centrality for networks produced by our approach is greater than that of either (or even both of) the dk- and BigV-produced networks, suggesting that a wider area of the space of solutions was explored. This holds for all three scenarios ($k = 5, 6, 7$). An important correlate of this finding is that neither BigV rewiring nor dk-decomposition can claim to generate null models. Interestingly, the networks produced by both methods do not appear to overlap suggesting that either methods generate networks in different areas of the space of solutions. Likewise, although our method appears to sample more widely than BigV rewiring and dk, full overlap only occurs for $k = 7$ whereas there is almost no overlap for $k = 5$. It remains to be seen whether, given more time, our method would uncover these areas of the space of solutions. Finally, given that the dk networks were produced from a single seed, it is worth pointing out that there was no obvious correlation between the betweenness centrality of the seed and the mean betweenness centrality for the dk-generated networks. The extent to which the choice of seed conditions the distribution of networks generated remains unclear.

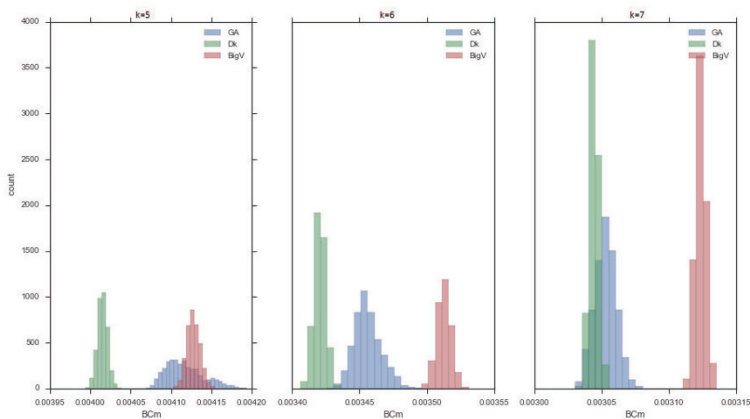


Fig. 4: Histograms of the mean betweenness centrality for the proposed method (blue), BigV rewiring (red) and dk2.1 (green) for each of the three scenarios: $k = 5$ (left), $k = 6$ (middle), $k = 7$ (right). The same number of networks was used for all three methods.

4 Discussion

In this paper, we have proposed a new GA-based approach to generating networks preserving degree distribution and global clustering. Our approach is focused on maximising the diversity of the networks being created. Since it is impossible to quantify the extent to which the entire space of solutions has been sampled, we have provided evidence of the effectiveness of the method by comparing it to two state of

the art network-generating methods, dk-series decomposition and BigV rewiring and showing that our method generates more diversity. Whereas coverage of the space of solutions using our method will depend on the number of generations available, both BigV rewiring and dk-series decomposition depend on a mixing time being reached. Care must therefore be taken in making definite statements about the ability of these methods to sample the range of networks found by our approach. However, given the same number of steps, there was greater diversity using our approach. This provides evidence for the usefulness of our method in the evaluation of the level of bias shown by current network generation methods. Much further work is needed to strengthen our framework, especially given that it is itself subject to a number of biases. For example, whilst encoding in terms of subgraphs provides much flexibility and scalability, it is itself a source of biases. At this time, it is unclear how a different choice of family would affect the diversity of networks uncovered. On the bright side, we believe that our starting scenario of networks with homogeneous distribution and low degree actually made it much harder to find diversity in the networks. The immediate focus will be to consider heterogeneous distributions with higher degrees. Whilst it will not affect computation time, it will provide much more flexibility for the network connection process (CMA) to realise networks (as well as remove the need to allow for 20% free edges, thus providing further control).

References

- [1] Aggarwal, C. C. (2011). An introduction to social network data analytics. In *Social network data analytics* (pp. 1-15). Springer US.
- [2] Colomer de Simón, P. (2014). RandNetGen [Computer software]. Retrieved from <https://polcolomer.github.io/RandNetGen/>. Last accessed 14 September 2016.
- [3] Danon, L., Ford, A.P., House, T., Jewell, C.P., Keeling, M.J., Roberts, G.O., & Vernon, M.C. (2011). Networks and the epidemiology of infectious disease. *Interdisciplinary perspectives on infectious diseases*, 2011.
- [4] Green, D. M., & Kiss, I. Z. (2010). Large-scale properties of clustered networks: Implications for disease dynamics. *J Biol Dyn*, 4(5), 431-445.
- [5] House, T., & Keeling, M.J. (2010). The impact of contact tracing in clustered populations. *PLoS Comput Biol*, 6(3), e1000721.
- [6] Karrer, B., & Newman, M.E. (2010). Random graphs containing arbitrary distributions of subgraphs. *Phys Rev E*, 82(6), 066118.
- [7] Keeling, M.J. (1999). The effects of local spatial structure on epidemiological invasions. *Proc R Soc Lond B: Biol Sci* 266(1421), 859867.
- [8] Kim, H., Toroczkai, Z., Erds, P.L., Miks, I., & Szekely, L.A. (2009). Degree-based graph construction. *J Phys A-Math Theor*, 42(39), 392001.
- [9] Kiss, I.Z., Miller, J.C., & Simon, P.L. (in Press). *Mathematics of epidemics on networks: From exact to approximate models*, Springer.
- [10] Klein-Hennig, H., & Hartmann, A. K. (2012). Bias in generation of random graphs. *Phys Rev E*, 85(2), 026101.
- [11] Lehman, J., & Stanley, K. O. (2011). Abandoning objectives: Evolution through the search for novelty alone. *Evol Comput*, 19(2), 189-223.
- [12] Mears, D., & Pollard, H. B. (2016). Network science and the human brain: Using graph theory to understand the brain and one of its hubs, the amygdala, in health and disease. *J Neurosci Res*, 94(6), 590-605.
- [13] Milo, R., Kashtan, N., Itzkovitz, S., Newman, M.E., & Alon, U. (2003). On the uniform generation of random graphs with prescribed degree sequences. arXiv preprint cond-mat/0312028.
- [14] Mouret, J.B., and Clune J. (2015). Illuminating search spaces by mapping elites. arXiv preprint arXiv:1504.04909.

- [15] Newman, M.E.J. *Networks: An introduction*. Oxford University Press, 2010.
- [16] Orsini, C., Dankulov, M.M., Colomer-de-Simón, P., Jamakovic, A., Mahadevan, P., Vahdat, A. & Fortunato, S. (2015). Quantifying randomness in real networks. *Nat Comm*, 6:8627.
- [17] Overbury, P., & Berthouze, L. (2015, July). Using novelty-biased GA to sample diversity in graphs satisfying constraints. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Lect Notes Comput Sc* (pp. 1445-1446). ACM.
- [18] Pastor-Satorras, R., Castellano, C., Van Mieghem, P., & Vespignani, A. (2015). Epidemic processes in complex networks. *Rev Mod Phys*, 87(3), 925.
- [19] Ritchie, M. and Berthouze, L. and Kiss, I.Z. (2016). Beyond clustering: Mean-field dynamics on networks with arbitrary subgraph composition. *J Math Biol* 72(1-2), 255-281.
- [20] Ritchie, M. and Berthouze, L. and Kiss, I.Z. (2016). Generation and analysis of networks with a prescribed degree sequence and subgraph family: higher-order structure matters. *J Complex Networks*, cnw011.
- [21] Stanley, K.O., & Miikkulainen, R. (2003). A taxonomy for artificial embryogeny. *Artif Life*, 9(2), 93-130.
- [22] Volz, E.M., Miller, J.C., Galvani, A., & Meyers, L.A. (2011). Effects of heterogeneous and clustered contact patterns on infectious disease dynamics. *PLoS Comput Biol*, 7(6), e1002042.

Within network learning on big graphs using secondary memory-based random walk kernels

Jianyi Lin, Marco Mesiti, Matteo Re and Giorgio Valentini

Abstract Significant advances in high-throughput sequencing technologies raised exponentially the rate of acquisition of novel biological knowledge in the last decade, thus resulting in consistent difficulties in the analysis of vast amount of biological data. This adverse scenario is exacerbated by serious scalability limitations affecting state-of-the art within-network learning methods and by the limited availability of primary memory in off-the-shelf desktop computers. In this contribution we present the application of a novel graph kernel, transductive and secondary memory-based network learning algorithm able to effectively tackle the aforementioned limitations. The proposed algorithm is then evaluated on a large (more than 200,000 vertices) biological network using ordinary off-the-shelf computers. To our knowledge this is the first time a graph kernel learning method is applied to a so large biological network.

1 Introduction

Many efforts have been devoted in the last decade to developing automated tools for large scale automated function prediction of proteins (AFP) [2, 3]. A recent international challenge for the critical assessment of automated function prediction [6], highlighted that scalability and heterogeneity of the available data represent two of the main challenges posed by AFP. From a learning perspective the problem is further complicated by the different functional annotation coverage in different organisms that make very difficult the effective transfer of the available functional knowledge from one organism to another. A possible approach for gene functional

Jianyi Lin (e-mail: jianyi.lin@kustar.ac.ae)
Khalifa University, Department of Applied Mathematics and Sciences, Abu Dhabi, United Arab Emirates,

Marco Mesiti (e-mail: marco.mesiti@unimi.it) · Matteo Re (e-mail: matteo.re@unimi.it) · Giorgio Valentini (e-mail: giorgio.valentini@unimi.it) ✉
Università degli Studi di Milano, Dip. di Informatica, Via Comelico 39/41 - 20135 Milano (MI)

annotation transfer between species relies on the availability of a collection of orthology relationships across interspecies proteins, and on the usage of an evolutionary relationships network as a suitable medium for transferring functional annotations to the proteins of poorly annotated organisms [7]. In this scenario a possible solution could be the application of network based learning methods on multi-species biological networks so that annotations coming from well annotated organisms could be used to effectively transfer functional annotations between species.

Unfortunately this approach is only apparently simple given the serious scalability limitations affecting graph-based learning algorithms (i.e. the popular label propagation and random walks based methods). These approaches usually rely on an in-memory adjacency matrix representation of the graph network, scale poorly with the size of the graph [9], and time complexity may become quickly prohibitive. When the size and structural complexity of the graph becomes so high that it is not possible to maintain it entirely in primary memory, alternative strategies (i.e. parallel/distributed computation [4, 11, 12], or secondary memory-based computation [5, 8, 19]) can be considered. However, at least in the case of the parallel/distributed computation, the identification of the optimal partitioning of the graph that minimizes the message passing requirements across a possibly large number of nodes of an HPC cluster is not immediate, especially in the case of very large and complex networks.

We previously proposed [13] a scalable semi-supervised network-based learning of protein functions algorithm that can be applied to large multi-species networks and is implemented using secondary memory-based technologies. Despite the appealing scalability performances from a learning standpoint this method is a classical random walk on graph. This paper extends the previous proposal by developing a local within network learning method based on the Random walk kernel [17] and a previously developed kernelized functions learning framework [15]. The novel local and secondary memory-based graph kernel method is compared with the classical random walks on graphs both in terms of learning performances and empirical time complexity.

To our knowledge this is the first reported case of application of a local and secondary-memory based graph kernel method to a very large biological network.

This manuscript is organized as follows. In the next section we introduce our proposed approach based on the local and secondary memory-based implementation of network-based algorithms (classical random walks on graph) for the multi-species AFP problem. We then present the novel random walks kernel local algorithm. We finally compare the algorithms in a multi-species AFP problem on a large biological network including 13 species of Eukaryotes and containing more than 200,000 proteins.

2 Local version of the classical Random Walk algorithm

Network-based algorithms learn by exploiting the overall topology of the networks [14, 15, 18], and their implementations usually require to process in primary memory a large part or the overall underlying graph. The main drawback of these implemen-

tations is that big networks cannot be entirely loaded into primary memory using off-the-shelf machines.

In [13] we developed local implementations of global network algorithms (classical random walks (RW) on graphs) by iteratively processing only one vertex and its incident edges at a time. In other words we do not reject to think globally by exploiting the overall topology of the network, but at the same time we solve locally by designing implementations of these algorithms through a vertex-centric programming model [4, 12].

A key feature of all the presented implementations is that the potentially very large matrices used by the primary-memory based versions of the classical random walk algorithm as well as of their kernelized versions are never computed nor stored entirely in main memory. All the learning algorithms presented in this paper were implemented by considering that:

- the existence of an edge in the network can be exploited as the only medium to propagate information between the vertices located at its endpoints;
- the knowledge of the set of edges originating from a vertex is enough to define its direct neighbourhood;
- the computation of the final score of a vertex can be decomposed in many steps each depending uniquely on the topology of the network and on the status of the vertices directly connected to the considered vertex;
- the score can be progressively accumulated into a single variable that is local to the vertices of the network.

RW algorithms [10] explore and exploit the topology of the functional network, starting and walking around from a subset $V_M \subset V$ of nodes belonging to a specific class M by using a transition probability matrix $\mathbf{Q} = \mathbf{D}^{-1}\mathbf{W}$, where \mathbf{D} is a diagonal matrix with diagonal elements $d_{ii} = \sum_j w_{ij}$. The elements q_{ij} of \mathbf{Q} represent the probability of a random step from i to j . The initial probability of belonging to M can be set to $p^o = 1/|V_M|$ for the nodes $i \in V_M$ and to $p^o = 0$ for the nodes $i \in V \setminus V_M$. If \mathbf{p}^t represents the probability vector of finding a “random walker” at step t in the nodes $i \in V$ (that is, p_i^t represents the probability for a random walk of reaching node i at step t), then the probability at step $t + 1$ is:

$$\mathbf{p}^{t+1} = \mathbf{Q}^T \mathbf{p}^t \quad (1)$$

and the update (1) is iterated until convergence. Given that a too deep exploration of the network can lead to a steady state with suboptimal learning performance, it is common practice to try with different number of predefined steps.

With the RW method at the steady state or at an optimized number of steps we can rank the vector \mathbf{p} to prioritize nodes according to their likelihood to belong to the class M under study.

Looking from a “local” perspective at RW algorithm, the update rule (1) becomes:

$$p_i^{t+1} = Q_i \cdot \mathbf{p}^t \quad (2)$$

where p_i is the probability of the i^{th} node, and Q_i represents the i^{th} column of the \mathbf{Q} probability transition matrix. By recalling that \mathbf{W} represents the original adjacency matrix of the graph and W_i its i^{th} column, from (2) we obtain:

$$p_i^{t+1} = D^{-1} W_i \cdot \mathbf{p}^t = \sum_{j=1}^n d_{jj}^{-1} w_{ji} p_j^t \quad (3)$$

This is the update rule of the random walk resolved at the i^{th} node of the graph, and can be viewed as a “local” version of (1): by updating all the nodes i of the graph, $1 \leq i \leq n$, we update the probability vector \mathbf{p}^{t+1} exactly in the same way of (1).

To compute (3) we need the following “local” data:

1. p_i^o (that is, the probability of the i^{th} node at start)
2. $d_{jj}^{-1} = \frac{1}{\sum_i w_{ji}}$ (that is, the sum of weights of the edges coming from j)
3. $w_{ji}, 1 \leq j \leq n$ (that is, the weights of the edges going to i)
4. $p_j^t, 1 \leq j \leq n$ (that is, the probabilities of nodes at the previous step).

If the graph is indirect (an this is the case for AFP problems), the weights of incoming and outgoing edges are the same, that is $\forall i, \forall j w_{ij} = w_{ji}$. This implies that we need to store only the list of edge weights outgoing from i : $L(i) = \{w_{ij} | w_{ij} > 0\}$. This in turn implies that in sparse graphs the spatial (and temporal) complexity at each node is sublinear. It is easy to see from (3) that the complexity of each iteration of the algorithm is $\mathcal{O}(n^2)$, but with sparse graphs, i.e. when $\forall i, |\{(j, i) | w_{ji} > 0\}| \ll n$, the complexity is $\mathcal{O}(n)$.

3 Local version of the Random walk kernel and Kernelized Score Functions

In [15] we proposed the kernelized score functions algorithmic framework that generalizes the notion of average, nearest neighbour and k-nearest neighbour distance from the set of positive nodes in a given network annotated to a specific functional class, and embeds a general kernel to model the functional similarity between nodes. This semi-supervised transductive learning method generalizes the guilt-by-association (GBA) approach [6] by introducing fast and efficient local learning strategies based on an extended notion of functional distance between the vertices, and adopts also a global learning strategy by using kernel functions able to exploit the relationships and the overall topology of the underlying network. The implementations presented in [15] were all “global” (primary memory-based). In order to significantly improve the scalability of the kernelized score functions we need to consider local implementations of:

1. The score function
2. The kernel embedded in the score function

The Average, Nearest Neighbour and k-nearest Neighbour score functions [15, 16] can be naturally implemented in “local” form once we are able to cast in local form the computation of the underlying kernel. In this work we focus on the Average score function:

$$S_{AV}(i, V_C) = \frac{1}{|V_C|} \sum_{j \in V_C} K(\mathbf{x}_i, \mathbf{x}_j) \quad (4)$$

where V_C is the set of positive vertices $i \in V$ that belongs to a given functional class and $\mathbf{x}_i, \mathbf{x}_j$ are features associated respectively with node i and j , usually represented as real vectors. To compute (4) we need the following “local” data:

1. The i^{th} row \mathbf{K}_i of the kernel matrix \mathbf{K}
2. The set V_C (indices of the positive columns)

The complexity is $\mathcal{O}(|V_C|)$, that is constant if $|V_C| \ll n$.

The “global” version of the 1-step random walk kernel is the following [17]:

$$\mathbf{K}_{rw} = (a-1)\mathbf{I} + \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}} \quad (5)$$

Where \mathbf{I} is the identity matrix, \mathbf{D} is a diagonal matrix with elements $d_{ii} = \sum_j w_{ij}$ and \mathbf{W} is the symmetric adjacency matrix of an indirect graph $G = (V, E)$. It is easy to derive from (5) the following “local” implementation of the 1-step random walk kernel, where $K(\mathbf{x}_i, \mathbf{x}_j)$, for the sake of simplicity is represented as k_{ij} :

$$k_{ij} = \begin{cases} d_{ii}^{-\frac{1}{2}} w_{ij} d_{jj}^{-\frac{1}{2}} & \text{if } i \neq j \\ (a-1) + d_{ii}^{-\frac{1}{2}} w_{ij} d_{jj}^{-\frac{1}{2}} & \text{if } i = j \end{cases} \quad (6)$$

To compute (6) we need the following “local” data for each edge (i, j) :

1. its weight w_{ij}
2. the values $d_{ii}^{-\frac{1}{2}}$ and $d_{jj}^{-\frac{1}{2}}$

The local computation complexity is constant. To compute the overall matrix the complexity is $\mathcal{O}(n^2)$. The q -step random walk kernel with $q > 1$ can be computed by following a step-by-step strategy based on this recursive formula: $\mathbf{K}_{rw}^q = \mathbf{K}_{rw}^{q-1} \mathbf{K}_{rw}$.

3.1 Putting together the Average score and the local version of the random walk kernel

By putting in (4) the local version of the random walk kernel (6), we obtain a “local” version of the *average score* with 1-step random walk kernel:

$$\begin{aligned} S_{AV}(i, V_C) &= \frac{1}{|V_C|} \sum_{j \in V_C} K(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{|V_C|} \sum_{j \in V_C} k_{ij} = \\ &= \frac{1}{|V_C|} \sum_{j \in V_C} \left(d_{ii}^{-\frac{1}{2}} w_{ij} d_{jj}^{-\frac{1}{2}} + \langle i = j \rangle (a-1) \right) \end{aligned} \quad (7)$$

where $\langle z \rangle$ is 1 if z is true and 0 otherwise. To compute (7) we need the following “local” data:

1. The row \mathbf{W}_i of the adjacency matrix \mathbf{W}
2. The set V_C (indices of the positive columns)
3. the values $d_{ii}^{-\frac{1}{2}}$ and $\{d_{jj}^{-\frac{1}{2}} | j \in V_C\}$

The local complexity is $\mathcal{O}(|V_C|)$, that is constant if $|V_C| \ll n$.

The main problem affecting the proposed solution for the local computation of the Average score based on a random walk kernel is that, using an approach starting from the adjacency matrix, a certain locality is maintained in the computation of the 1-step and 2-steps but when we compute RWK with 3 or more steps we need the overall matrix and the “locality” is completely lost.

To overcome the complexity problems raising from the local computation of the p -step RWK with $p > 1$, we propose an iterative version of the kernelized score functions with random walk kernels.

3.2 Iterative computation of the kernelized Average Score function with p -step RWK

As stated in Section 3.1 the iterative nature of the p -step RWK computation poses serious challenges from a local implementation perspective and a progressive locality loss make solutions based on simple modification of the classical random walks unsuitable for real world big graph. In order to overcome this limitation we propose a novel representation of the combined RWK-kernelized score functions computation that better fits the constraints imposed by the analysis of very large graphs and by the vertex-centric programming paradigm.

More precisely, we propose an iterative formula for computing the Average kernel score with a p -step random walk for the whole graph. Such formula consists in a simple matrix-vector multiplication at each iteration.

Recall that for every node $i \in V$ the average score of the p -step random walk kernel starting from $V_C \subset V$ is $S_{AV}(i, V_C) = \frac{1}{|V_C|} \sum_{j \in V_C} (K^p)_{ij}$. Let's denote the column vector constructed from $S_{AV}(i, V_C)$ by varying i with

$$S_{AV}(V_C) = [S_{AV}(1, V_C), \dots, S_{AV}(n, V_C)]^T.$$

It can be shown that the vector $S_{AV}(V_C)$ of average scores for the whole graph $G = \langle V, E \rangle$ with a p -step random walk kernel starting from nodes of $V_C \subset V$ can be computed as $S_{AV}(V_C) = \mathbf{D}^{\frac{1}{2}} \mathbf{v}^p$ by the iterative formula

$$\mathbf{v}^p = \mathbf{M} \mathbf{v}^{p-1} \quad \text{where } \mathbf{M} = [(a-1)\mathbf{I} + \mathbf{D}^{-1}\mathbf{W}]$$

with the initialization vector \mathbf{v}^0 having element

$$v_i^0 = \begin{cases} \frac{1}{|V_C| \sqrt{d_{ii}}} & \text{if } i \in V_C; \\ 0 & \text{otherwise.} \end{cases}$$

We will now show how to compute the average score for the p -step RWK using a local implementation, i.e. on a vertex-based graph computation model.

Consider the “global” iterative formula $\mathbf{v}^p = \mathbf{M} \mathbf{v}^{p-1}$; denote by v_i^p the i -th element of \mathbf{v}^p and by $I(i) = \{j \in V : w_{ij} > 0\}$ the incoming neighbors of i in the weighted graph G . We have

$$v_i^p = \sum_{j \in V} M_{ij} v_j^{p-1} = \sum_{j: M_{ij} > 0} M_{ij} v_j^{p-1}.$$

Since

$$M_{ij} = \begin{cases} a - 1 & \text{if } i = j \\ \frac{w_{ij}}{d_{ii}} & \text{otherwise.} \end{cases}$$

we can establish the rule for updating the value of a vertex i in the graph:

$$v_i^p = \sum_{j \neq i: w_{ij} > 0} \frac{w_{ij}}{d_{ii}} v_j^{p-1} + (a - 1)v_i^{p-1} = d_{ii}^{-1} \sum_{j \in I(i)} w_{ij} v_j^{p-1} + (a - 1)v_i^{p-1}. \quad (8)$$

Finally, using the iteratively computed v_i^p value, the vertex-centric score S_{AV} can be easily obtained:

$$S_{AV}(i, V_C) = d_{ii}^{\frac{1}{2}} v_i^p \quad (9)$$

Therefore, in a vertex-based graph computation model, at every update step it is sufficient to take into account for each vertex $i \in V$:

- the old value v_j^{p-1} of all neighboring vertices $j \in I(i)$,
- the weight w_{ij} of all incoming edges,
- the weighted in-degree d_{ii}

and then use the previous rule for the updating. This kind of computation scheme can be implemented in any vertex-based graph analytics programming framework.

4 Experimental settings

We applied our methods based on the local implementation of network-based algorithms and secondary memory-based computation to the multi-species protein function prediction in eukarya. In all the experiments we implemented the network-based methods using *GraphChi*, a software library for large-scale graph computation using secondary memory [8]. All the experiments have been performed using off-the-shelf desktop computers with a limited amount of RAM memory (4 GB). It is worth noting that in these experimental conditions random-walk algorithms that store in primary memory the adjacency matrix of the graph described in Section 4.1 run out-of-memory due to the limited amount of available RAM.

In the remainder of this section we summarize the experimental set-up and the characteristics of the data, and then we compare the empirical computational time and the performance of secondary memory-based implementations of network learning algorithms for AFP.

4.1 Dataset description and performance evaluation

In order to test the ability of the proposed local methods to scale to large multi-species networks, we constructed a large genes network (hereafter referred to as Eukarya-net). All the proteins interactions composing Eukarya-net were downloaded in pre-computed form from the STRING protein-protein interactions database. STRING (<http://string-db.org/>) is a collection of networks composed by real and predicted gene-gene interactions (based on genetic data, physical data and literature

data) and aims at providing a global view of all the available interaction data, including lower-quality data and/or computational predictions for as many organisms as feasible. Starting from the STRING interaction data (version 9.05), we selected all the Eukaryotic species having 10,000 or more proteins. The selected Eukaryotic species are listed in Table 1.

As class labels for the proteins included in Eukarya-net we used the Gene ontology [1] (GO) annotations available in STRING (version 9.05). The STRING website provides flat text files containing a mapping from GO annotations to STRING proteins and a STRING internal confidence score for each GO annotation, ranging from 1 (low confidence) to 5 (high confidence). While extracting the GO labels we considered only the annotations with confidence score 5. We then filtered out all the GO terms associated with less than 20 and more than 100 proteins (473 GO terms). We finally randomly selected from this set 50 GO terms.

Performance were evaluated in terms of runtime, Area under the Receiver Operating curve (AUROC), and Precision at fixed Recall levels using a canonical 5-fold stratified cross validation scheme.

4.2 Results

Table 2 summarizes the average per-term runtime required to complete a 5-fold cross validation with the Eukarya-net involving more than 200,000 proteins of 13 multi-cellular eukarya organisms.

Table 1: Selected species from the core region of the STRING protein networks database

NCBI taxon ID.	Species	n. proteins
3218	<i>Physcomitrella patens</i>	10352
3702	<i>Arabidopsis thaliana</i>	23576
7227	<i>Drosophila melanogaster</i>	12845
7739	<i>Branchiostoma floridae</i>	16418
8364	<i>Xenopus (Silurana) tropicalis</i>	13678
9031	<i>Gallus gallus</i>	13119
9258	<i>Ornithorhynchus anatinus</i>	13333
9606	<i>Homo sapiens</i>	20140
9615	<i>Canis lupus familiaris</i>	16912
10090	<i>Mus musculus</i>	20023
13616	<i>Monodelphis domestica</i>	15409
39947	<i>Oryza sativa Japonica</i>	13330
69293	<i>Gasterosteus aculeatus</i>	13307

Table 2: Average per-term empirical time complexity between the compared local and secondary memory-based network learning methods implementations

Local algorithm	per-term empirical time complexity evaluation
1-step RW	21.46s
2-step RW	33.19s
3-step RW	46.69s
1-step RWK (Average score)	21.05s
2-step RWK (Average score)	34.05s
3-step RWK (Average score)	46.25s

We observe that the average computational time is very similar for both the RW and RWK-based kernelized functions secondary memory-based implementations. The performance (see Table 3) in terms of the average precision at fixed recall levels obtained in this test are relatively low, especially when compared with the high average AUC obtained with the RW at 1, 2 and 3 steps. The observed relatively low precision can be explained by taking into account that it is more negatively affected by class unbalance and, in the Eukarya-net network task, the positives are at most 100 while the number of vertices in the network is 202,442 (i.e. the positives are less than 0.05% of the vertices at best). Note that in this case the 2-steps RW achieves the best AUROC results: it is likely that these results could be due to the connections between nodes representing proteins coming from different species but further evaluation is required in order to clarify the observed results.

Table 3: Average AUC, precision at 20% recall (P20R) and precision at 40% recall of the compared local and secondary memory-based network learning methods across 50 GO terms. Performance estimated through 5-fold cross-validation.

Algorithm	AUROC	P20R	P40R
RW - 1 step	0.8601	0.1449	0.0943
RW - 2 steps	0.9667	0.1329	0.0929
RW - 3 steps	0.9598	0.0927	0.0785
RWK 1 step (Average score)	0.9106	0.2115	0.1422
RWK 2 steps (Average score)	0.9902	0.2670	0.1605
RWK - 3 steps (Average score)	0.9680	0.2314	0.1498

As we can see the best performances in terms of AUROC are obtained at two steps also with the RWK-based average score function. While the differences in AUROC performances between the classical and kernelized random walks based methods are not so big, the same does not hold with respect to the performances in terms of P20R and P40R (respectively precision at 20% and 40% recall), where the random walk kernel clearly outperforms the local classical random walk-based gene function predictor.

Overall, these results show that the secondary memory-based implementation of kernelized score functions allow the analysis of big networks using off-the-shelf desktop computers, and achieve results competitive with the classical random walk algorithm in the multi-species prediction of protein functions.

5 Conclusions

In this work we presented a novel secondary memory-based implementation of a Random walk kernel network learning method. More precisely, we developed a novel local and secondary memory-based algorithm able to compute a kernelized score function (the average score) embedding a p -step random walk kernel. The proposed algorithm has been applied to the prediction of protein functions in the context of a multi-species large biological network involving more than 200,000 proteins. The experimental results show that the local version of the kernelized version of the random walk exhibits an empirical time complexity comparable with a local RW and secondary memory-based within network learning algorithm, and outperforms the classical random walk algorithm for the multi-species prediction of protein functions. From a more general standpoint we believe that “local” versions of network-based algorithms, together with an efficient secondary memory-based implementation, can open new avenues for the analysis of big and complex networks in computational biology, without the mandatory need of complex clusters of computers or expensive stand-alone workstations equipped with very large RAM memory.

References

- [1] M. Ashburner et al. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1), 2000.
- [2] I. Friedberg. Automated protein function prediction-the genomic challenge. *Brief Bioinform.*, 7:225–242, 2006.
- [3] J. Gillis and P. Pavlidis. Characterizing the state of the art in the computational assignment of gene function: lessons from the first critical assessment of functional annotation (cafa). *BMC Bioinformatics*, 14(3):S15–10, 2013.
- [4] J.E. Gonzalez et al. Powergraph: Distributed graph-parallel computation on natural graphs. In *Proc. of the 10th USENIX Conf. on Operating Systems Design and Implementation*, pages 17–30, 2012.
- [5] W.S. Han et al. Turbograph: a fast parallel graph engine handling billion-scale graphs in a single PC. In *Proc. of the 19th ACM SIGKDD Int’l Conf. on Knowledge Discovery and Data Mining*, pages 77–85, 2013.

- [6] Y. Jiang et al. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biology*, 17(184), 2016.
- [7] A. Kuzniar et al. The quest for orthologs: finding the corresponding gene across genomes. *Trends Genet.*, 24(11):539–551, 2008.
- [8] A. Kyrola et al. Graphchi: large-scale graph computation on just a pc. In *Proceedings of the 10th USENIX Conf. on Operating Systems Design and Implementation*, pages 31–46, 2012.
- [9] W. Liu, J. Wang, and S.F. Chang. Robust and scalable graph-based semisupervised learning. In *Proc. IEEE*, volume 100, pages 2624–2638, 2012.
- [10] L. Lovasz. Random Walks on Graphs: a Survey. *Combinatorics, Paul Erdos is Eighty*, 2:1–46, 1993.
- [11] Y. Low et al. Graphlab: a new parallel framework for machine learning. In *Conf. on Uncertainty in Artificial Intelligence (UAI)*, 2010.
- [12] G. Malewicz et al. Pregel: a system for large-scale graph processing. In *Proc. of the ACM SIGMOD Int'l Conf. on Management of Data*, pages 135–146, 2010.
- [13] M. Mesiti, M. Re, and G. Valentini. Think globally and solve locally: secondary memory-based network learning for automated multi-species function prediction. *GigaScience*, 3(1):1, 2014.
- [14] S. Mostafavi et al. GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome Biology*, 9(S4), 2008.
- [15] M. Re, M. Mesiti, and G. Valentini. A Fast Ranking Algorithm for Predicting Gene Functions in Biomolecular Networks. *IEEE ACM Transactions on Computational Biology and Bioinformatics*, 9(6):1812–1818, 2012.
- [16] M. Re and G. Valentini. Cancer module genes ranking using kernelized score functions. *BMC Bioinformatics*, 13(Suppl 14/S3), 2012.
- [17] A.J. Smola and I.R. Kondor. Kernel and regularization on graphs. In *Proc. of the Annual Conf. on Computational Learning Theory*, LNCS, pages 144–158. Springer, 2003.
- [18] G. Valentini, G. Armano, M. Frasca, J. Lin, M. Mesiti, and M. Re. RANKS: a flexible tool for node label ranking and classification in biological networks. *Bioinformatics*, 32:2872–2874, 2016.
- [19] J. Webber et al. A programmatic introduction to neo4j. In *Proc. of the 3rd Annual Conf. on Systems, Programming, and Applications: Software for Humanity*, pages 217–218, 2012.

A Method for Evaluating the Navigability of Recommendation Algorithms

Daniel Lamprecht, Markus Strohmaier and Denis Helic

Abstract Recommendations are increasingly used to support and enable discovery, browsing and exploration of large item collections, especially when no clear classification of items exists. Yet, the suitability of a recommendation algorithm to support these use cases cannot be comprehensively evaluated by any evaluation measures proposed so far. In this paper, we propose a method to expand the repertoire of existing recommendation evaluation techniques with a method to evaluate the navigability of recommendation algorithms. The proposed method combines approaches from network science and information retrieval and evaluates navigability by simulating three different models of information seeking scenarios and measuring the success rates. We show the feasibility of our method by applying it to four non-personalized recommendation algorithms on three datasets and also illustrate its applicability to personalized algorithms. Our work expands the arsenal of evaluation techniques for recommendation algorithms, extends from a one-click-based evaluation towards multi-click analysis and presents a general, comprehensive method to evaluating navigability of arbitrary recommendation algorithms.

1 Introduction

Websites with large collections of items need to support three ways of information retrieval: (i) retrieval of familiar items (ii) retrieval of items that cannot be explicitly described but will be recognized once retrieved and (iii) serendipitous discovery [30]. For a website with a large collection of items, such as an e-commerce website, (i) can be enabled with a full-text search function. For (ii) and (iii), however, a search function is generally not sufficient. These types of information retrieval are therefore

Daniel Lamprecht (e-mail: daniel.lamprecht@tugraz.at) · Denis Helic (e-mail: dhelic@tugraz.at)
KTI, Graz University of Technology

Markus Strohmaier (e-mail: strohmaier@uni-koblenz.de)
GESIS and University of Koblenz-Landau

often supported by recommendations that connect items and enable discovery and navigation.

Users have been found to enjoy perusing item collections such as e-commerce sites or recommender systems without the immediate intention of making a purchase [14]. More generally, some users prefer navigation to direct search even when they know the target [29]. For platforms where users immediately consume content, such as YouTube or Quora, recommendations serve the use case of *unarticulated want*, and are therefore a crucial part of the user experience [10]. In item collections that do not associate descriptions or metadata with content (such as videos) frequently no clear structuring of items exists, and recommendations play a vital role in the user interfaces. It is therefore critical for these systems to support discovery via links.

When a website provides recommendations along with each item, the items and the associated recommendations form a *recommendation network*—an implicit view of a recommender system where items are nodes and recommendations are edges. This type of recommendations are frequent on e-commerce websites, such as Amazon (“customers who bought this also bought”). Many websites associate a fixed number of recommendations with each item, which leads to a constant outdegree and a varying indegree for each node in the network

Knowing more about recommendation networks would give web-site operators the possibility to assess the effects of recommendations and help to produce recommendations that make it easier for users to discover and explore items. While a few studies have already looked at recommendation networks and provided first important insights into the nature and structure of these networks [6, 8, 19, 28], there is no systematic approach to evaluating the navigability of recommendation algorithms.

This paper presents a general method to evaluate the practical navigability of arbitrary recommendation networks by using simulations based on three navigation models established in the literature, namely *point-to-point navigation* [15], *navigation via berrypicking* [2] and *navigation via information foraging* [27]. The combination of established techniques from the fields of network science and information retrieval allows us to present a novel method that extends common evaluation measures towards a path-based evaluation and expands the arsenal of existing recommendation evaluation techniques.

We show the feasibility of this method by applying it to four non-personalized recommendation algorithms on three datasets and investigate their properties. We also illustrate the general suitability of our method to personalized recommendations and report initial results for a sample configuration.

2 Related Work

Initially, recommender systems were mostly evaluated in terms of prediction accuracy [11]. However, the focus on accuracy has been found to neglect other important applications of recommender systems such as support for the discovery of novel items, browsing, or diversified recommendations, and may lead to a bias towards popular items [8] or a filter bubble effect [24]. For these reasons, a series of evaluation

metrics for additional properties of recommender systems has been developed. These metrics include diversity [4, 7], novelty [7, 11], serendipity and coverage [11, 14] and are considered orthogonal to prediction accuracy.

The evaluation method presented in this paper is rooted in Stanley Milgram's small world experiments [23], which laid the foundation for *decentralized search*. Kleinberg [16] and Watts [32] later formalized the property that a navigable network requires short paths between all (or almost all) nodes. Kleinberg also found that an *efficiently navigable* network possesses certain structural properties that make it possible to design efficient decentralized search algorithms that only have local knowledge of the network [15]. The delivery time of such algorithms is then sub-linear in the number of network nodes. In this paper, we investigate the efficient navigability of recommendation networks through the simulation of navigation models based on decentralized search.

The static topology of recommendation networks has been extensively studied for the case of music recommenders [8, 28]. Their corresponding recommendation networks have been found to exhibit heavy-tail degree distributions and small-world properties [6], implying that they are efficiently navigable with local search algorithms. A first study [19] has already explored the reachability and navigability of the recommender systems of IMDb. The corresponding recommendation networks were shown to lack support for navigation scenarios. However, the use of diversified recommendations was able to substantially improve this and lead to more navigable recommendation networks. A similar methodology has been applied to suggest links to improve navigability on Wikipedia [18].

3 Evaluation Method

Navigation is at the core of exploration and browsing, which are important use cases of a recommender system, as many users find browsing pleasant [14], use it to discover novel content [21] or consume the content along the browsing path (e.g., on YouTube). A defining property of online navigation is that the knowledge about a website is mostly local: users only perceive the links emanating from the current page and generally only have intuitions about where those links might lead, but lack global knowledge about the system. In the case of a top-N recommender system, users are generally only aware of the recommendations with the current item.

The evaluation method we propose makes use of greedy decentralized search to simulate navigation in recommender systems and measures the success rate. This model has been used in previous work to analyze navigation dynamics in networks [12, 13] and has been found to produce comparable results to human navigation patterns [20, 31]. At each step, this algorithm evaluates a heuristic for every present link and greedily selects the one maximizing that heuristic. We take the heuristic to represent vague intuitions about navigation that users might gain from looking at the descriptions of recommendation targets. For example, if a user was looking for a new science-fiction movie, they might be tempted to follow recommendations to other science fiction movies based on the title, a brief textual description or the displayed

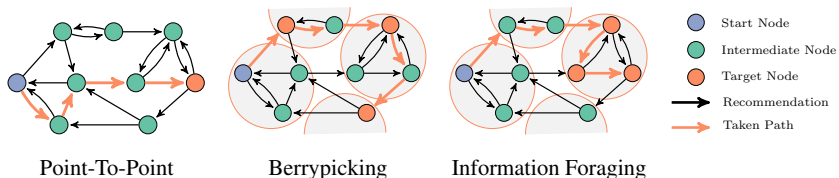


Fig. 1: **Information Seeking Scenarios.** We use three information seeking scenarios to study navigability of recommendation networks. The objective in point-to-point navigation is to find a single goal item. For berrypicking, we cluster the networks and set the goal of finding any one item in four clusters (shown in gray). For information foraging, the goal is to find multiple items in a single cluster.

image. We use an implementation that does not revisit previously explored nodes. In case no unvisited item is present, the simulation backtracks.

A number of information seeking models have been established in the literature. To investigate the general suitability of recommendation algorithms to navigation based on different approaches, we evaluate navigation scenarios based on three of these models: point-to-point navigation [15], berrypicking [2], and information foraging [27]. For all scenarios, the start and target nodes in the network are determined independently of the network structure, i.e., regardless of whether the recommendation algorithm actually enabled a path between them. This allows us to fairly compare all recommendation algorithms and shows how well they support navigability. In what follows, we describe the three navigation scenarios in more detail (cf. Figure 1).

Point-To-Point Navigation Point-to-point navigation [15] represents the task of finding a single target item in a recommendation network and models the navigational behavior of users with a specific item in mind that they cannot explicitly describe. For example, a user could try to find a science-fiction movie with a specific motif or to rediscover something on tip of their tongue. As such, this scenario covers point (ii) (“retrieval of items that cannot be explicitly described”) of Toms’s ways of information retrieval [30]. We then simulate navigation starting at the start node of a pair and with the objective of reaching the target node. As start-target pairs we sample pairs of nodes proportionally to how often they were corated by users in the corresponding rating dataset.

Navigation via Berrypicking Berrypicking is an information seeking model which regards information seeking as a dynamic process where the information need is evolving and can be satisfied by multiple pieces of information in a *bit-at-a-time retrieval*—an analogy to picking berries on bushes [2]. Berrypicking can be thought of as covering points (ii) (“retrieval of items that cannot be explicitly described”) and (iii) (“serendipitous discovery”) of Toms’s ways of information retrieval [30]. We model this scenario based on clusters, which we obtain with k -means based on the rating vectors. We randomly pick a first cluster and then draw one of the top four closest clusters based on Euclidian distance randomly. We then repeat this to

find two more clusters. Starting from a randomly chosen node in the first cluster, the objective of the scenario is then to reach any node from the second cluster, followed by any node from the third and then the fourth cluster. In this way, the scenario models the evolving stages of berrypicking, where users inspect an item and adapt their information needs based on it.

Navigation via Information Foraging Information foraging [27] is an information seeking theory inspired by optimal foraging theory in nature, where organisms have adopted strategies maximizing energy intake. For instance, when foraging on a patch of food, an animal must decide when to move on to the next patch (e.g., when finding apples on a tree is becoming too tedious). Some of the same mechanisms have identified for human information seeking behavior, where humans try to maximize information gain. Information can be modeled as occurring in patches, and information seekers as guided by *information scent* [9]. In a scenario based on information foraging, we model the scenario of depleting a patch of information. We assume that the objective is to retrieve nodes in a patch—guided by information scent in terms of the search heuristic. We take information foraging to model points (ii) and (iii) (“retrieval of items that cannot be explicitly described”) and “serendipitous discovery”) of Toms’s ways of information retrieval [30].

Baselines We evaluate two baseline solutions: An *optimal solution* uses the shortest possible paths (that users with perfect knowledge of the network could take). A *random solution* performs a random walk with no background knowledge at all.

4 Experimental Setup

We use three datasets for this paper:

- **MovieLens** is a film recommender systems maintained by GroupLens Research at the University of Minnesota. For this work, we use their dataset consisting of one million ratings from 6,000 users on 4,000 movies.
- **BookCrossing** is a book exchange platform. For this work, we use a 2005 crawl of the website [33]. We use only the explicit ratings, combine ratings for duplicate books and use ratings from users with ≥ 20 ratings on ≥ 5 books. This leaves us with roughly 50,000 ratings by 1,088 users on 3,637 books.
- **IMDb** is a database of movies and TV shows. We use a 2015 crawl of the website [19], from which we use ratings for items published in 2013 and 2014 and condense them in the same way as for the BookCrossing dataset, resulting in 2.3M ratings for 6,690 titles by 37,216 users.

We calculate recommendations in the following way: For a given set of items I and a recommendation algorithm R , we use R to compute the pairwise similarities for all pairs of items $(i, j) \in I$. For each item $i \in I$, we then define the set of the top- N most similar items to i as $L_{i,N}$. We then create a directed top- N recommendation network $G(V, N, E)$, where $V = I$, N is the number of recommendations available for each

item and $E = \{(i, j) \mid i \in I, j \in L_{i,N}\}$. This method leads to recommendation networks with constant outdegree and varying indegree—representing a typical setting.

For simplicity's sake, we investigate recommendation algorithms based on non-personalized recommendations. The similarities these recommendations are based on, however, are directly taken from the similarities used in the personalized recommendation algorithms. They therefore represent the recommendation networks as an unregistered or newly registered user would see them. For most websites, the vast majority of visitors does not contribute or register—this is known as the *90-9-1 Rule* (90% lurkers, 9% intermittent contributors and 1% heavy contributors) [25, 26]. However, our method is general and also applicable to personalized recommendation algorithms, which we exemplarily demonstrate in Section 6.

We use the following four recommendation algorithms in this work:

Association Rules (AR) Association rules are based on the market-basket model, where, in this case, we put all items rated by the same user into a basket and regard ratings as binary (i.e., rated/not rated). For every ordered pair of items (i, j) , we then rank all items by how much more likely an item is to be consumed after a given item was consumed (similar to the Apriori algorithm [1]). Specifically, we compute the fraction of co-ratings of i and j over the total ratings of i (i.e., the fraction users who rated both i and j , out of those who rated i). Let U_i be the set of users who rated item i . We can then compute this as $(|U_i \cap U_j|)/(|U_i|)$. To compensate for the popularity of j , we then divide by the fraction of users who did not rate i but still rated j . Let \bar{U}_i be the set of users who did not rate item i . We can then divide by $(|\bar{U}_i \cap U_j|)/(|\bar{U}_i|)$ to counter the effect of highly popular items that are likely to be co-rated with every item, but would not be very useful as a recommendation. We then take the top- N items most likely to be co-rated with it.

Collaborative Filtering (CF) For a given user u and an unrated item i , item-based collaborative filtering predicts the rating of u for i from a small number of other items that u previously rated. These other items are commonly selected as the ones maximizing the centered cosine similarity to i . The rating prediction is then computed as the weighted sum of their ratings, weighted by their similarity. To obtain unpersonalized recommendations, we compute the centered cosine similarity of an item i to all other items j in the dataset and use the top- N .

Interpolation Weights (IW) Interpolation weights are computed in a similar way to item-based collaborative filtering. However, instead of using a predefined similarity measure (such as the centered cosine similarity) to weight the contributions of other ratings, *interpolation weights* representing the relations between pairs of items are learned from the data. We use gradient descent to learn item-based interpolation weights by minimizing the root-mean square error for predictions on a test set [3] and then use the resulting weights as the similarity measure to obtain the top- N most similar items to an item.

Matrix Factorization (MF) Matrix factorization describes both items and users of a recommender system by affinities to a number of latent factors [17]. To find these factors, this algorithm factorizes the rating matrix U into two matrices as $U = Q^T P$

that represent the associations of users and items with the latent factors. We learn these matrices by minimizing the root-mean-square prediction error on a test set with gradient descent. After this minimization, we represent each item by the vector of its association with the latent factors and compute the centered cosine similarity between the latent factors for all pairs of items to obtain the top- N most similar items.

As the heuristic for decentralized search, we use the TF-IDF cosine similarity of brief textual descriptions of titles (namely title and plot summary of IMDb for the movies and the summary provided by GoodReads for the books). At each step, the simulation uses this heuristic to select the link leading to the item that has the highest TF-IDF cosine similarity to the navigation goal. We use a heuristic independent of ratings to decouple it from the recommendations used to generate the networks. For sake of brevity, we only report the results for a deterministic greedy search with 50 steps. However, we also evaluated all simulations for 10 and 25 steps as well as with an ϵ -greedy approach [12] and found that, while the total success rates decreased, the relative differences between the approaches did not change.

We evaluate a total of 1,200 navigation simulations per scenario. For the clusters, we only use those consisting of 4–30 nodes to balance the difficulty. The target of the navigation for the berrypicking and information foraging scenario is represented by the centroid of the target cluster. The TF-IDF cosine similarity of a potential link target l is therefore represented by the average of the similarity between l and all items in the target cluster.

5 Results

Point-To-Point Navigation The first row of Figure 2 displays the success rate (i.e., the fraction of successful simulations) for point-to-point navigation. Since the number of steps per simulation (50) is larger than the distances between all start-target pairs in the recommendation networks, the optimal solutions (shown in gray bars) correspond to all start-target pairs between which a path of any length existed. The optimal solution is therefore a measure of how well a recommendation algorithm theoretically supports this navigation scenario. The second baseline approach is a random walk, which shows the success rates achievable by an uninformed random process and serves to demonstrate that the simulations based on greedy search are able to exploit the link selection heuristic to reach navigation goals. The simulation for point-to-point navigation with greedy search for $N = 5$ recommendations leads to an average success rate of 6.86%. This indicates that users would be able to retrieve only a very small share of items in the recommender systems by focused point-to-point navigation. For $N = 20$ recommendations, the success rates increase substantially (average of 24.4%). Recommendations generated by interpolation weights lead to the best success rates (42–48%).

Navigation via Berrypicking For five recommendations, the success rates for the case of genre-based clusters are 14.5% on average. With 20 recommendations, this

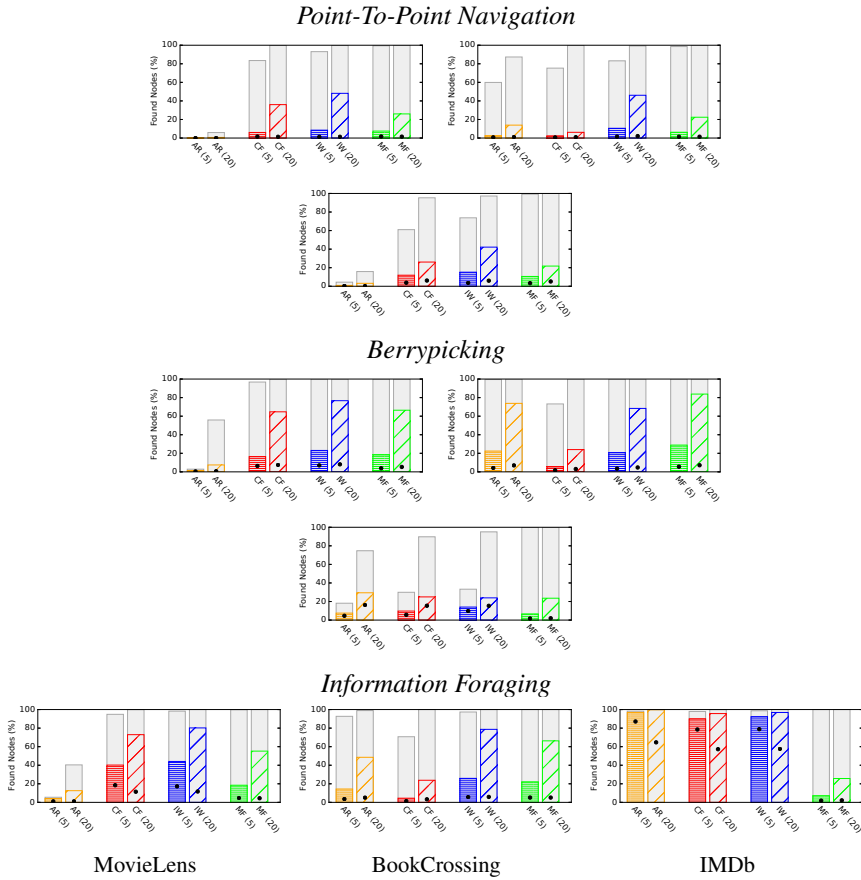


Fig. 2: Success Ratios for the navigation simulations. The bars depict the average percentage of found targets. Baseline success rates are depicted as gray bars (optimal solutions) and black dots (random walk solutions). Success rates are computed as the average number of found targets. Recommendation networks generated by interpolation weights (IW) generally performed best.

increases to to 47%. Since the targets consists of three clusters, a success rate of 33% indicates that an average of one cluster was found.

The success rates for the IMDb dataset are substantially lower than for the other two datasets. A more detailed analysis shows that the networks for IMDb are clustered more strongly than those of the other two datasets. For a dynamic information seeking scenario such as berrypicking, this means that the simulation of adapting information needs was not very well supported for IMDb. Overall, recommendations generated by matrix factorization and interpolation weights fared best.

Navigation via Information Foraging A priori, it is not clear if retrieving multiple items from the same cluster represents an easier task than retrieving them from

different clusters, as a cluster of items does not necessarily mean that items are located in proximity in the recommendation network. However, the resulting success rates show that items from the same clusters in the network are easier to retrieve: five recommendations lead to a success rate of 38.3%, and twenty recommendations to 63.1%. This indicates that the recommendation algorithms are able to use the characteristics in the ratings to support both genre-based and rating-based clustering.

The success rates again measures the number of found items in a cluster. The results for this scenario show that the success rates for the baselines, namely the random walks and the optimal solutions are consistently very high, which also indicates that the network structures reflect the clustering very well. Whereas for berrypicking, the simulations on the IMDb dataset perform poorly, the contrary is the case for information foraging, where the success rates range up to 99%. This again confirms the strong clustering in these networks, that lead to densely interconnected regions among similar items and facilitate retrieval in the same cluster. Recommendations generated with interpolation weights generally fare best.

6 Personalized Recommendations

We now demonstrate the general suitability of our method to personalized recommendation approaches and report initial results for a sample configuration of parameters. The key difference for personalized recommendations is that a separate recommendation network emerges for every user based on their rated items. For this illustration, we follow the approach of Amazon.com, as detailed by Linden, Smith and York in 2003 [22], which consists of two steps: First, a set of similar items is determined for each item. Second, the items with the highest predicted rating among this set are recommended. We study two variants of this:

- **Pure.** We first compute a candidate set of similar items for an item—these are simply the non-personalized recommendations. Then we select the N items from this set that have the highest predicted rating for the specific user.
- **Mixed.** We again compute the set of similar items, but only use the $N/2$ recommendations with the highest predictions and add the $N/2$ top non-personalized recommendations (without introducing duplicates).

For both algorithms, we allow the recommendation of items that the user had already rated (which is yet another parameter to tune). We note that for this setting, the differences between the personalized networks for users decrease. When not allowing this, the resulting recommendation networks show a decrease in navigability the more items a user has already rated. For sake of space, we only report results for a restricted set of parameters. The results for the other combinations of parameters were similar, but we leave it to future work to examine them in more details.

Figure 3 shows the evaluation for recommendations generated by interpolation weights and matrix factorization for the user with the median number of ratings in the BookCrossing dataset. The outcome is generally similar to non-personalized networks. The pure algorithm leads to notably higher success rates for the optimal

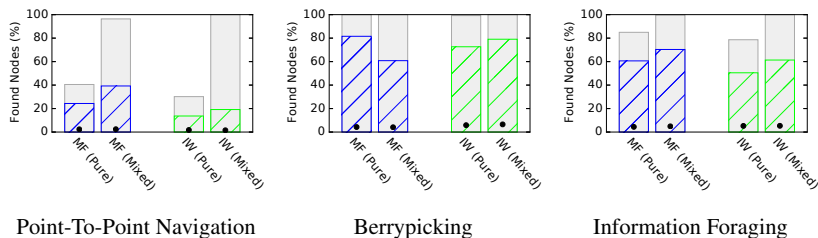


Fig. 3: **Navigational Success Rates for Personalized Recommendations.** All simulations were evaluated for BookCrossing, 20 recommendations and personalized for the user with the median number of ratings in the dataset. The results show that while the mixed recommendations enable a better optimal solution, the recommendations did not reflect the intuitions of the navigation simulations very well.

solution, but not for the simulation results themselves. This indicates that while the mixed algorithm leads to a better connectivity in the networks, this was not necessarily the case for navigability. This in turn suggests that the recommendations generated by this algorithm did not capture the intuitions used in the navigation simulations very well. In future work, the evaluation method proposed in this paper could be used to develop a more effective personalized recommendation selections.

7 Discussion

We have presented a novel evaluation method that expands the repertoire of recommendation evaluation measures with a technique to assess navigability. The proposed method evaluates the navigation dynamics of recommendation networks by simulating three different navigation models, namely point-to-point navigation, navigation via berrypicking and navigation via information foraging. We believe that applying this method can broaden our understanding of recommendation algorithms and lead to a more complete characterization of their properties. In practice, this method could be used to improve the experiences of users as they navigate recommendation networks (such as recommended videos on YouTube).

To demonstrate the feasibility of our method, we applied it to three exemplary datasets and highlighted differences in navigability for four different, non-personalized, recommendation algorithms. For five recommendations per item, we find that the recommendation algorithms we investigate considerably limit the navigability. However, we find that it can be improved by raising the number of recommendations. For the three navigation scenarios we investigate we find that the explorative scenarios inspired by berrypicking and information foraging lead to the best retrieval performance, while the scenario based on point-to-point navigation was less well supported. While increasing the number of recommendations represents a simple solution, a large number of recommendations could potentially clutter the interface and overwhelm users [5]. This shows that there is still a substantial potential

to improve recommendation algorithms to better support navigation dynamics. As for the recommendation algorithms, we find that the recommendations generated by interpolation weights and matrix factorization performed best overall. However, more work is necessary to confirm these findings.

The selection of algorithms and datasets was naturally arbitrary, but they serve the purpose of illustrating the evaluation and therefore do not limit our main contribution of presenting a novel evaluation method. We have shown the suitability of our method for non-personalized recommendation algorithms and thereby effectively inspected recommendation networks for users who are either new to the system or simply browsing without being registered, and have also illustrated the applicability of our method to personalized recommendations.

The navigation models applied in this method are well-established in the research community and cover a wide range of typical user interaction scenarios with information systems in general, and recommender systems in particular. Greedy decentralized search, the basis for our navigation scenarios based on these models, has been used in previous work to analyze navigation dynamics in networks [12, 13] and has been found to produce comparable results to human navigation patterns [20, 31]. The navigation models we used do, however, have limitations and were deliberately kept simple, as the focus of our work was not on the information seeking models and their validity but on the properties of the recommendation algorithms. However, this does not limit our work, as our evaluation method does not depend on this particular model, which can easily be adapted or exchanged in future work. Possible enhancements to the navigation models could include a teleportation element (as in PageRank) modeling jumps between items without recommendations.

In summary, our work extends common evaluation measures of recommendation algorithms towards a path-based evaluation. Just as the evaluation of recommender systems has been shifting from accuracy-based measures towards diversification, coverage and time-dependent evaluations, we believe that our method helps to push the frontier of recommendation algorithms towards producing recommendations that make it easier for users to discover and explore items.

Acknowledgements This research was supported by a grant from the Austrian Science Fund (FWF) [P24866].

References

- [1] Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: VLDB'94 (1994)
- [2] Bates, M.J.: The design of browsing and berrypicking techniques for the online search interface. *Online Information Review* **13**(5), 407–424 (1989)
- [3] Bell, R.M., Koren, Y.: Improved neighborhood-based collaborative filtering. In: KDD Cup and Workshop at the KDD'07 (2007)
- [4] Boim, R., Milo, T., Novgorodov, S.: Diversification and refinement in collaborative filtering recommender. In: CIKM'11 (2011)

- [5] Bollen, D., Knijnenburg, B.P., Willemsen, M.C., Graus, M.: Understanding choice overload in recommender systems. In: *RecSys'04*. ACM (2010)
- [6] Cano, P., Celma, O., Koppenberger, M., Buldú, J.M.: Topology of music recommendation networks. *Chaos* **16**(1) (2006)
- [7] Castells, P., Hurley, N.J., Vargas, S.: Novelty and diversity in recommender systems. In: *Recommender Systems Handbook*, pp. 881–918. Springer (2015)
- [8] Celma, O., Herrera, P.: A new approach to evaluating novel recommendations. In: *RecSys'08* (2008)
- [9] Chi, E.H., Pirolli, P., Chen, K., Pitkow, J.: Using information scent to model user information needs and actions on the web. In: *CHI'01* (2001)
- [10] Davidson, J., Liebal, B., Liu, J., Nandy, P., Vleet, T.V., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., Sampath, D.: The youtube video recommendation system. In: *RecSys'10* (2010)
- [11] Gunawardana, A., Shani, G.: Evaluating recommender systems. In: *Recommender Systems Handbook*, pp. 265–308. Springer (2015)
- [12] Helic, D., Strohmaier, M., Granitzer, M., Scherer, R.: Models of human navigation in information networks based on decentralized search. In: *HT'13* (2013)
- [13] Helic, D., Strohmaier, M., Trattner, C., Muhr, M., Lerman, K.: Pragmatic evaluation of folksonomies. In: *WWW'11* (2011)
- [14] Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* **22**(1), 5–53 (2004)
- [15] Kleinberg, J.M.: Navigation in a small world. *Nature* **406**(6798), 845 (2000)
- [16] Kleinberg, J.M.: The small-world phenomenon: An algorithmic perspective. In: *TOC'00* (2000)
- [17] Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer* **42**(8), 30–37 (2009)
- [18] Lamprecht, D., Dimitrov, D., Helic, D., Strohmaier, M.: Evaluating and improving navigability of wikipedia: A comparative study of eight language editions. In: *OpenSym'16* (2016)
- [19] Lamprecht, D., Geigl, F., Karas, T., Walk, S., Helic, D., Strohmaier, M.: Improving recommender system navigability through diversification: A case study of IMDb. In: *IKNOW'15* (2015)
- [20] Lamprecht, D., Strohmaier, M., Helic, D., Nyulas, C., Tudorache, T., Noy, N.F., Musen, M.A.: Using ontologies to model human navigation behavior in information networks: A study based on wikipedia. *Semantic Web* **6**(4), 403–422 (2015)
- [21] Lerman, K., Jones, L.: Social Browsing on Flickr. In: *ICWSM'07* (2007)
- [22] Linden, G., Smith, B., York, J.: Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing* **7**(1), 76–80 (2003)
- [23] Milgram, S.: The small world problem. *Psychology Today* **1**(2), 60–67 (1967)
- [24] Nguyen, T.T., Hui, P.M., Harper, M.F., Terveen, L., Konstan, J.A.: Exploring the filter bubble: the effect of using recommender systems on content diversity. In: *WWW'14* (2014)
- [25] Nielsen, J.: The 90-9-1 rule for participation inequality in social media and online communities (2006). www.nngroup.com/articles/participation-inequality
- [26] Nonnecke, B., Preece, J.: Lurker demographics: Counting the silent. In: *CHI'00* (2000)
- [27] Pirolli, P.: *Information Foraging Theory: Adaptive Interaction with Information*. Oxford University Press (2007)
- [28] Seyerlehner, K., Knees, P., Schnitzer, D., Widmer, G.: Browsing music recommendation networks. In: *ISMIR'09* (2009)
- [29] Teevan, J., Alvarado, C., Ackerman, M.S., Karger, D.R.: The perfect search engine is not enough: A study of orienteering behavior in directed search. In: *CHI'04* (2004)
- [30] Toms, E.G.: Serendipitous information retrieval. In: *DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries* (2000)
- [31] Trattner, C., Singer, P., Helic, D., Strohmaier, M.: Exploring the differences and similarities between hierarchical decentralized search and human navigation in information networks. In: *IKNOW'12* (2012)

- [32] Watts, D.J., Dodds, P.S., Newman, M.: Identity and search in social networks. *Science* **296**, 1302–1305 (2002)
- [33] Ziegler, C.N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: WWW'05 (2005)

Part III
Community Structure

A New Decision Technique For Sub-community And Multi-Level Knowledge Extraction In Social Networks

Joseph Ndong and Ibrahima Gueye

Abstract A suitable state model can be retrieved from a Karhunen-Loeve Transformation to build a new decision process from which, we can extract useful knowledge and information about the identified underlying sub-communities from an initial network. The aim of this method is to build a framework for a multi-level knowledge retrieval. So, besides the capacity of this methodology to reduce the high dimensionality of the data, the new detection scheme is able to extract, from the sub-communities, the most relevant nodes and the dense sub-groups with the definition and formulation of new quantities related to the notions of energy and co-energy. The energy of a node is the rate of its participation on a the whole set of activities while the notion of co-energy defines the rate of interaction/link between two nodes. These two important features are used to make each link weighted and bounded, so that we will be able to perform a thorough refinement of the sub-community discovery. This study allows to perform a multi-level analysis by extracting information either per-link or per-intra-sub-community. This methodology is applied to a real world dataset where the workload of activities over a set of events is considered.

Key words: Social network analysis, community detection, PCA, KLT, Energy;

1 Introduction

The paper focuses on sub-community detection with, as main ideas, the reduction of the dimensionality of the dataset in order to maintain the only relevant part of the data. This can be achieved by analyzing the correlation of the data features. We consider a stochastic process where temporal and/or spatial correlations might happen in the features of the data being delivered. It is possible that, to build a framework for the

Joseph Ndong (e-mail: josephndong@ucad.edu.sn)✉
University Cheikh Anta Diop, Dakar, Senegal

Ibrahima Gueye (e-mail: igueye@ept.sn)
Polytechnic School, Thies, Senegal

scope of sub-community identification in a social network, the data in interest might be collected in the same location at the same time or in different distant areas at the same period of time. So, temporal correlations involve in situations where events, which depend to each other, happen in the same time period in a given location while spatial correlations appear when the events happen in different locations or geographic areas at the same time. If two (or more) variables are correlated in time and/or space, the behavior of an actor should remain unchanged if we merge the relevant information of the variables. Taking into account all the variables independently would be out of interest to learn the interactions between actors and to find their potential relationships. So, the fact of reducing the dimension of the data should have a positive impact in the reduction of the complexity of the study.

PCA is a powerful tool to find relevant patterns in data of high dimension. Despite its strength, it has been proven that PCA is very sensitive to its parameter settings. It was also used extensively with the assumption of linearity and sufficiency of mean and variance. In the past decade, it has been shown that very bad results were often obtained since this assumption is not valid [11]. Here, we propose a more elaborated development of this technique with a robust extension known as the Karhunen-Loeve (KL) transform [4]. In the best of our knowledge, this work is the first study related to social network analysis which makes a thorough emphasis of the stochastic nature of the process under consideration. The temporal correlation that might govern this process is not favorable to use the classical principal component tool for the scope of dimensionality reduction. Instead, we propose a more elaborate method based on a Karhunen-Loeve transform. Nevertheless, some studies, based on PCA, have already built techniques for sub-community identification. For instance, [2] has proposed the use of PCA to extract the meaningful variables over a huge number of data features from the popular Youtube network. The authors of [10] have also provided a technique based on PCA for the purpose of variables selection in order to reduce the dimension of the dataset. In [1], the authors performed a principal component analysis of the rankings produced by 39 existing and proposed measures of scholarly impact that were calculated on the basis of both citation and usage log data, in order to learn about the impact of scientific publications in terms of citation counts. In all those studies, the PCA technique were applied in its original form, so it suffers to the problems we have mentioned about of lack of linearity and sufficiency of mean and variance. In the present work, we develop a more robust method where the PCA approach is suited to perform with stochastic processes and to build a suitable decision variable for the identification of communities from the initial network.

2 Contribution of this work

This work is dedicated to the implementation of an extended version of the classical principal component analysis tool, namely the Karhunen-Loeve Transformation (KLT), in order to build a state model from which we can retrieve a suitable decision variable for the scope of establishing a convenient algorithm for the purpose of sub-community detection and analysis of the intra dynamics of the system. The method

presents several advantages: (i) the possibility to build sub-communities of level α from an initial large network, (ii) each link between two nodes in a sub-community is weighted with a rate in $[0; 1]$ to quantify the intensity of the relation between two actors, (iii) each actor will be identify with a weight in $[0; 1]$ corresponding to the degree of his participation to the activities in the network, (iv) after detection a given sub-community, since each link is weighted, we can identify dense intra sub-community (i.e. a group of linked actors which have the same weight/energy), (v) whenever a link is detected between two nodes, we determine the "qualifier of each node". The "qualifier" is a label we attribute to a node; its value is either "superior" or "inferior". To derive this property, we evaluate two useful information. First, we would like to know, between two linked nodes, which one influences much more the other. The response is that, the node with the highest energy within a detected link can be consider as the "superior" and the other the "inferior". If the two nodes have the same energy, we call them "twins nodes". Second, the probability of existence of the link can be bounded in order to know how much energy is necessary, at least, to maintain the link over time. Whenever the energy of one node is less than the inferior limit of the bounded interval, the link will disappear.

We will give, in the following, all the definitions and formulations around the notion of energy and bounded link and show how to technically achieve our objectives.

3 Methodology and algorithm for the detection

The purpose of the work is to build a methodology for sub-community tracking and detection based on the analysis of a huge number of features corresponding to events/activities for which a group of actors/nodes participate. First, we aim at finding the main features, to incorporate in our model, by means of extended principal component analysis. The second relevant issue of this work is related to the specification of a new detection procedure consisting of merging all the relevant features into a single process we will label as a "Decision Variable" (DV). By analyzing this process for the sub-community tracking operation, we can discover subgroups of actors using a multi-level thresholding and the notion of "energy dissipation" of an actor over the events.

We consider a community of R actors $\Omega = (a_1, \dots, a_R)$ which perform activities on a set of K initial correlated events (e_1, \dots, e_K) . For each event e_k , we have a column vector of size R containing the amount of participation of all R actors to the corresponding activity. This operation gives us the $R \times K$ matrix of correlated random variables $X = (X_1, \dots, X_K)$. On other words, one observes these random variables through R independent realization vectors $x^i = (x_1^i, \dots, x_K^i)$ $i = 1, \dots, R$. After extracting the relevant components from the KL transformation, we can build our decision variable as a row vector $DV = (y_1, \dots, y_K)$. At this point, we can set a certain number of concepts for our methodology. We introduce the notion of "energy dissipation" (Ed) to quantify the degree of importance a given actor put on a series of events. This notion is simple and intuitive. When considering the set of events/activities, the events for which the actor puts a high degree of importance

constitutes his energy. For example, we can consider money as energy. When one goes to buy some products at the market, we can say that he/she is dissipating a certain amount of his/her energy. In this case, one should buy a "product A", and consequently buy another "product B" necessary to use the product A. Here, we can see the notion of correlation between these products/variables. When an athlete performs several disciplinary exercises in sport, we can view his actions as the dissipation of his energy over the different events, in order to win a medal. The energy of an actor is thus quantifiable, its a measure of the strength of his participation the a series of activities.

If the actor participates actively to all or most of the activities which a high probability, then his energy increases, otherwise we say that this actor has less energy according to the ensemble of events happening at a given period of time.

Recall that the DV variable contains the aggregated amount of all actors participation to all events. So, the energy dissipation Ed of actor i is the row vector defined as:

$$Ed_i = \left\{ k, /x_k^i \geq DV[k], \forall k = 1, \dots, K \right\} \quad (1)$$

Ed_i contains all the index of events for which the energy dissipation is greater than the reference DV. Consequently, we can calculate the total energy of the actor i as the real value:

$$E_i = \frac{|Ed_i|}{|DV|} \quad (2)$$

where $|\cdot|$ indicates the size of a vector.

We will also refer to the notion of "co-energy" dissipation (CED) as the amount of energy between two actors according to their participation to the same set of activities. This quantity is a measure of the mean energy produced simultaneously by the two actors on the same activities:

$$CED_{ij} = \frac{|(Ed_i \cap Ed_j)|}{|DV|} \quad (3)$$

Finally, our detection procedure boils down to fixing a threshold α and to put a link between actor i and actor j if the rate of their co-energy exceeds the limit α . This means that the following inequality must hold to add the link:

$$CED_{ij} \geq \alpha \quad (4)$$

When Eq. 4 holds, hence, the value of CED_{ij} becomes the weight of the link between actor i and actor j . And then, this link is bound by the interval $[\min(E_i, E_j), \max(E_i, E_j)]$. By varying the threshold $\alpha \in [0; 1]$, one can build many different sub-communities with the same dataset, each sub-community with a score α which measures its degree of realization. The algorithm to achieve our aim is described as follow:

4 From PCA to its Karhunen-Loeve Transform Expansion

The principle of principal component analysis consists at observing a set of random variables $X = (X_1, \dots, X_K)$ and to seeking for the most suitable non-canonical

Algorithm 6 *Sub-Community Discovering*

Input: C_t , a community
 $\Omega(C_t)$, the sets of actors within C_t
 $x^i = (x_1^i, \dots, x_K^i)$ the vector of participation of actor i
 $DV = (dv_1, \dots, dv_K)$ the decision variable

1: α , the link detection threshold

Output: V , a sub-community

2: Calculate Co-Energy dissipation between actors and apply threshold to add link

3: **Begin**

4: **for all** $(k, l) \in \Omega(C_t), k \neq l$ **do**

5: Apply Eq. (1)

6: $Ed_k = \left\{ p, /x_p^k \geq dv_p, \forall p = 1, \dots, K \right\}$

7: $Ed_l = \left\{ p, /x_p^l \geq dv_p, \forall p = 1, \dots, K \right\}$

8: Apply Eq. (3)

9: $CED_{kl} = \frac{|(Ed_k \cap Ed_l)|}{|DV|}$

10: **end for** ▷ Apply threshold to decide to put a link, Eq. (4)

11: **if** $CED_{kl} \geq \alpha$ **then**

12: $addLink(V, k, l)$

13: **end if**

14: Return V

15: **End**

basis (e_1, \dots, e_K) to represent the random variables X . By assuming linearity and sufficiency of mean and variance, the most suitable basis is the one for which the variance is maximized for each projected component. This basis is then (ϕ_1, \dots, ϕ_K) , where ϕ_i is an eigenvector of the covariance matrix of X defined by the quantity $\mathbb{E}\{(X - \mu)(X - \mu)^T\}$, where μ is the column vector containing the means of X_i . The eigenvectors can be retrieved by the equation:

$$\sum \phi_i = \lambda_i \phi_i \quad (5)$$

where λ_i are the eigenvalues of the above covariance matrix. As the covariance matrix is positive definite, the resolution of the Eq. 5 gives at most K positive eigenvalues and K orthogonal eigenvectors. By performing the singular value decomposition (SVD) on the covariance matrix, we have the basis change matrix $U = [\phi_1, \dots, \phi_K]$ which contains the eigenvectors ϕ_i . After applying PCA, one can easily rewrite the initial vector of random variables X in the new coordinate system as:

$$X = \sum_{i=1}^K Y_i \phi_i \quad (6)$$

where Y_i are jointly independent random variables with mean 0 and variance λ_i . PCA replaces the random variables X by a vector of independent random variables Y that are linearly equivalent. When the dataset under consideration is not in contradiction with the conditions of mean and variance sufficiency and linearity, applying PCA

can be a convenient way to reduce to dimensionality of the data. The SVD procedure can be easily performed with the estimated covariance matrix $\frac{1}{N-1}xx^T$ to find the basis change matrix. When the linearity is not guaranteed, using such an orthogonal basis can result to erroneous interpretation. So we propose an extension of PCA to stochastic processes.

We consider our sample of dataset $X(t = (X_1(t), \dots, X_K(t))^T$ as stochastic processes that have temporal dependencies, with a covariance function $\sigma_{i,j}(\tau) = \mathbb{E}X_i(t)X_j(t - \tau)$ defined over an interval $[a, b]$. So the Karuhen-Loeve theorem states that we can rewrite the vector as a series expansion as follow:

$$X_l(t) = \sum_{i=1}^K \sum_{j=1}^{\infty} Y_{i,j}^l \Phi_{i,j}(t), \tag{7}$$

where $Y_{i,j}^l$ are pairwise independent random variables and $\Phi_{i,j}(t)$ are pairwise orthogonal deterministic (non-random) functions defined on $[a, b]$, i.e.: $\int_a^b \Phi_{i,j}(t) \Phi_{m,n}^*(t) dt = 0$, for $i \neq m$ or $j \neq n$. Generally, the basis functions $\Phi_{i,j}(t)$ are re-scaled such that $\int_a^b |\Phi_{i,j}|^2(s) ds = 1$.

This theorem extends PCA to a vector of stochastic processes as Eq. 7 is the equivalent of Eq. 6. The family of deterministic functions $\Phi_{i,j}(t)$ is an orthogonal basis for the space of linear stochastic processes and the random variables $Y_{i,j}^l$ are coordinates of the stochastic process $X_l(t)$ in this new space. We can formally derive the basis functions $\Phi_{i,j}(t)$ by solving the following set of linear integral equations:

$$\sum_{i=1}^K \int_a^b \sigma_{i,l}(s) \Phi_{i,j}(s-t) ds = \lambda_{l,j} \Phi_{l,j}(t), j > 0. \tag{8}$$

This set of equations is the equivalent of Eq. 5. The random variables $Y_{i,j}^l$ are obtained by projecting each stochastic process over an eigenfunction:

$$Y_{i,j}^l = \int_a^b X_l(s) \Phi_{i,j}(s) ds \tag{9}$$

The KL expansion considers the temporal correlation between time t and $t +$ as well as the spatial correlation between process $X_i(\cdot)$ and $X_j(\cdot)$. This results in a more complex analysis than the simple PCA described earlier. However, this higher complexity is unavoidable because of the temporal correlation. Not taking it into account leads to the errors described in [9].

The Galerkin method [7] can be used to truncate the KL expansion to N terms. This operation transforms the above integral equations to a matrix problem that can be solved by applying the SVD technique. This makes it possible to derive the KL expansion using only a finite number of samples. The Galerkin method generates a set of eigenvectors in a $K \times N$ dimensional vector space, that are time-sampled versions of the originally continuous function $\Phi_{i,j}(t)$. Finally, we obtain a discrete version of the KL expansion as:

$$X_l[k] = \sum_{i=1}^K \sum_{j=1}^N Y_{i,j}^l \Phi_{i,j}[k] \tag{10}$$

We first have to estimate the spatio-temporal correlation matrix. To do so, we construct a $KN \times (nN)$ observation matrix:

$$x = \begin{pmatrix} x_1(1) & \dots & x_1(n-N) \\ x_1(1) & \dots & x_1(n-N+1) \\ \vdots & \ddots & \vdots \\ x_1(N) & \dots & x_1(n) \\ x_2(1) & \dots & x_2(n-N) \\ \vdots & \ddots & \vdots \\ x_2(N) & \dots & x_2(n) \\ \vdots & \ddots & \vdots \\ x_K(1) & \dots & x_K(n-N) \\ \vdots & \ddots & \vdots \\ x_K(N) & \dots & x_K(n) \end{pmatrix} \tag{11}$$

The matrix

$$\hat{\Sigma} = \frac{1}{n-N-1} x^T x \tag{12}$$

contains all the needed spatio-temporal covariance estimates. The Galerkin method consists of applying PCA to this large matrix. This results in KN eigenvectors $\Phi_{i,j}[\cdot]$ of length KN that are used to construct a basis transform matrix U . The coefficients $Y_{i,j}^l$ are obtained by applying the basis change transform $y = Ux$. Applying KL expansion to K stochastic processes entails diagonalizing a $KN \times KN$ matrix (in place of a $K \times K$ matrix in the traditional PCA).

Now if we neglect some of the smaller terms of the expansion (terms with small values of $\text{Var}\{Y_{i,j}^l\}$) we obtain a linear approximation of the initial process in a smaller dimension vector space. The discrete expansion in Eq. 10 is therefore approximated as:

$$\hat{X}_l(kT) = \sum_{i=1}^L \sum_{j=1}^M Y_{i,j}^l \Phi_{i,j}^k, \tag{13}$$

where $M < N$ and $L < K$. This approximation has a noteworthy optimality property. Among all approximations defined over a linear space of dimension LM , this is the linear approximation with the smallest approximation error variance $\text{Var}\{X(t) - \hat{X}(t)\}$. The basis change transform becomes a $KN \times LM$ matrix U_{LM} that contains the LM eigenfunctions $\Phi_{i,j}[\cdot]$ in its columns. This is the theoretical basis to use the KL expansion as a non-parametric and generic technique for modeling a large class of processes where we cannot reject the linearity and sufficiency of mean and variance.

The expansion in Eq. 13 provides a synthesis method for generating an approximated process $\hat{X}_l[k]$ by a bank of ML filters with Finite Impulse Response equal to $\Phi_{i,j}[k], k = 0, \dots, KN$; each filter being excited by the random variable input $Y_{i,j}^l$. By predicting the values of the realization of the KN random variables $Y_{i,j}^l$ by applying the basis change matrix to observation $X[\cdot]$, we can use this synthesis filter as a model to build a decision variable suited to analyze a given community for the purpose of sub-community detection.

5 Building the Decision Variable

After applying the KL transformation, we retrieve easily the relevant principal components (i.e. the components with highest eigenvalues) by means of the scree test of Cattell [3]. Thereafter, we assume that the K linear stochastic processes in vector $X[k]$ are linear processes, i.e., one can represent them using a dynamic state space representation as $Z[k+1] = AZ[k] + e[k]$, where $Z[k]$ is a KN dimension matrix constructed by concatenating N vectors ($X[k], \dots, X[kN]$) and $e[k]$ is a vector of KN independent and identically distributed (iid) random variables. In this work, $Z[k]$ contains the amount of activities performed by the N actors for the K events. It is interesting to see that the relation between $Z[k]$ and $Z[k+1]$ in the quantity A can be interpreted as the influence an event has on another event with respect to the activity of the actor on that two events. For example, if $A = 1$, we can say that the rate of participation of an actor to an event numbered $k+1$ is nearly equal to the number of involvement of the same actor on an event k . The random variable $e[k]$ takes into account the error one could do to say that the given two events might be correlated in case where in reality there is no correlation. Now assuming that the process vector $X[k]$ is approximated by a finite KL expansion with LM terms, there is therefore a U_{LM} basis transform matrix that maps $Z[k]$ into the new coordinate: $\zeta_R[k] = U_{LM}\hat{Z}[k]$ ($\zeta_R[k]$ being the reduced coordinate vector of dimension LM). The inverse projection can be found through $\hat{Z}[k] = U_{LM}^T\zeta_R[k]$. The Maximum Likelihood framework [11] can be used successfully to derive an approximation of the process $\hat{Z}[k]$ as: $\hat{Z}[k] = U_{LM}U_{LM}^T Z[k]$. A one-dimensional decision variable DV is finally derived as a function of the estimated multi-dimensional process $\hat{Z}[k]$ as follow:

$$DV[k] = \left(\frac{Q[k]}{\phi_1} \right)^{h_0} \quad (14)$$

where $Q[k] = \sum \hat{Z}[k]$ (summation of all the raws to merge the whole amount of actor's participation on the events) and $h_0 = 1 - \frac{2\phi_1\phi_3}{3\phi_2^2}$, $\phi_i = \sum_{j=r+1}^m \lambda_j^i$; for $i = 1, 2, 3$. Jensen et al. [6] give an approximation of this variable by a gaussian distribution of mean $1 + \phi_2 h_0 (h_0 - 1) / \phi_1^2$ and variance $2\phi_2 h_0^2 / \phi_1^2$.

In [11], the authors have built their decision variable as a function of the prediction error $e[k] = Z[k] - \hat{Z}[k]$ and have set $Q[k] = e[k]^T e[k]$. This choice was suitable for the scope of anomaly detection to tracking anomalous events which might appear as volume anomalies attacks. The amount of volume anomalies in communication networks can be positive or negative frequencies. In our work, we can't use the same definition since we manipulate the amount of actor's activities which is always quantified as positive values in \mathcal{N} . Another reason which demonstrate that the use of the error prediction is inappropriate is that $e[k]$, in favorable conditions, should be a zero mean process; whenever $Z[k]$ and $\hat{Z}[k]$ are in accordance, $e[k]$ would be equal to zero and thus can't quantify the amount of actor's activities. So, the convenient variable able to merge (using the summation $Q[k] = \sum \hat{Z}[k]$) and quantify the amount of participations of all actors to the activities is the estimated process itself, i.e. the matrix $\hat{Z}[k]$.

6 Validation

We validate our approach on the real world collection of data coming from **Reddit.com** [5]. We use several samples of different sizes and, build four scenarios A, B, C and D with dimension $(N \times K, N$ the number of actors and K the number of events) $10 \times 15, 10 \times 150, 10 \times 500$ and 10×1200 respectively. In Table .1, we give an idea on the content of the data, in each column vector, we have the total amount of submissions to an image by the set of actors.

Actors \ events	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_{15}
1	11	0	11	4	0	2	0	4	18	2	0	6	16	1	0
2	5	0	0	0	0	0	1	0	0	0	2	0	2	0	0
3	1	0	2	0	3	1	1	0	1	0	0	2	1	0	2
4	4	1	0	0	0	1	2	0	7	0	1	0	9	1	0
5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0
7	0	2	0	0	1	0	0	0	0	0	0	4	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 1: Activities and amount of actor participation to submissions on events. Scenario A.

Two levels of information can be retrieved from the results. In the first level, we have the results about the formation of the underlying sub-communities. This result corresponds to the natural clustering of the different nodes according to the energy provided by each of them. The second level of information refers to the characteristics of links and nodes inside the given sub-groups. This refinement provides useful information when one wants to emphasize and explore some parts of the network.

6.1 Information about the formation of sub-communities

One of the main objectives of social network analysis is related to clustering in order to study the similarities inside the network [8, 12]. So, the first result is about the formation of sub-communities. The graphs in Fig. 1 show all the groups we discover with the different scenarios. In Fig. 1a, we see a sparse sub-community within two dense sub-communities. The term 'sparse' refers to a group of nodes with different levels of energy. When we observe a group of linked nodes with the same value for their energy, we consider this group as an inner dense sub-community; here, we have $\{1, 3, 7\}$ and $\{2, 3, 7\}$.

In graphs of Fig. 2, we draw the co-energy participation between two nodes to emphasise the fact the role of the Decision Variable DV have to set a potential link. Considering all the events at the same time, a link can be put between two nodes if the amount of participations of both two nodes, for the same set of events, exceeds

by far the reference point given by the decision variable. We use circles to identify the events where the energy of the actors is higher than the reference point. Clearly, for most of the given events, if the energy of each node exceeds the value of the DV for that events, then we put a link between the two nodes.

6.2 Information about the intrinsic behavior inside sub-communities

The second level of information this technique might deliver is about the dynamics of nodes and their relation inside the detected sub-groups. So, another result is related to the boundary of each detected link. In Table 2, we have for each node, its total energy, i.e. the probability of this node to participate to all events. By observing carefully this table and the graphs of the sub-communities, we see that the bounds of a link is the interval $[a, b]$, where a and b are the respective total energy of the specified nodes. And so, the score/weight of a link is always inside this interval, as we can observe for all links detected. For example, in Fig. 1d, the link between node #4 (with energy 0.68) and node #7 (with energy 0.74) has a weight of 0.59 and its bounded interval is $[0.68, 0.75]$. As the network evolves, whenever the energy of a node belongs out of the interval, the link will disappear. By inspecting frequently the evolution of the bounded interval, one can retrieve useful information about the degree of importance of the different nodes by analyzing their energy.

Each of the other scenarios (B, C and D) give also a sparse sub-community.

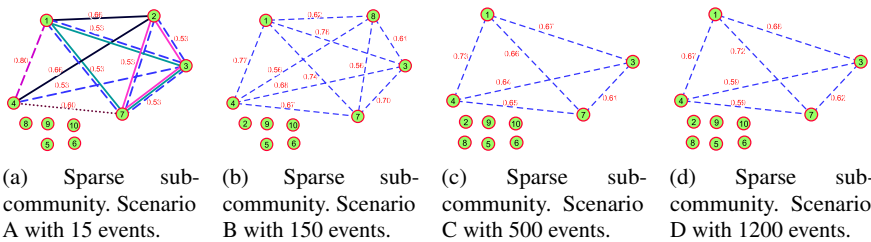


Fig. 1: Results of the sub-community members identification. In scenario A 1a, we can observe two dense sub-communities. In the other scenarios (B, C and D), we obtain only a sparse sub-community. The different scenarios are built with different size of the vector of events.

7 Conclusion

In this work, we have developed a new technique related to an extended version of principal component analysis to build a methodology for the purpose of community detection in a social network. This technique is more elaborated to run within

Table 2: Total Energy of each actor for the different scenarios.

Energy of the different actors (Eq. 3)											
	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	Number of events
E_i	0.80	0.66	0.53	1	0.26	0.40	0.60	0	0	0	15
E_i	0.88	0.25	0.86	0.77	0.46	0.25	0.74	0.62	0.15	0.13	150
E_i	0.79	0.26	0.79	0.74	0.23	0.19	0.69	0.46	0.11	0.06	500
E_i	0.86	0.27	0.70	0.68	0.25	0.23	0.74	0.31	0.97	0.55	1200

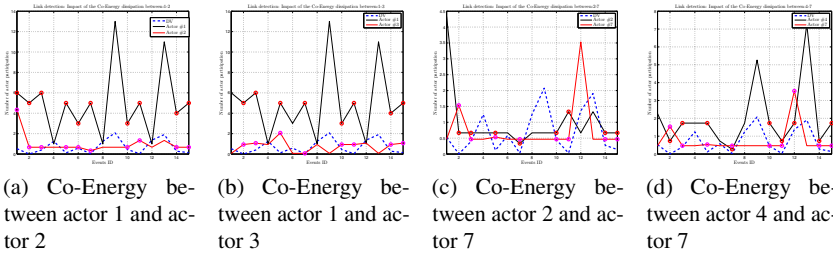


Fig. 2: Impact of the Co-Energy dissipation between two actors in the link detection phase. The circles represents events where both the two actors have their co-energy higher than the reference value in the decision variable DV. These actors are linked since their co-energy dissipation concern more than half of the events.

stochastic process than the classical PCA which is designed originally to solve the problem of dimensionality reduction for univariate dataset. The main innovation of this work is manifold: (i) we define the notion of co-energy between two nodes to quantify the intensity of their relation, (ii) we can also extract the proper energy of a given node to know how it influences the overall community, (iii) technically, the KL-PCA technique makes possible to build a decision variable and to form a state model from which we apply a decision process to identify each link. The introduction of the notion of energy make possible to see potential intra sub-communities (i.e. nodes with the same co-energy) inside a sub-community; (iv) each detected link is bounded, so we know how much energy is necessary to maintain a link over time. As a perspective, we plan to learn more the impact of the energy of the nodes. Clearly, it would be important to know how the amount of energy of given (selected) nodes should influence the behavior of the community by maintaining this community stable/unchanged over time or broken up. It would be interesting also to study the impact of the energy of each node or a set of nodes with high or less energy on the behavior of the entire network while the network grows in terms of new nodes and/or the arrival of new data.

References

- [1] Bollen, J., Van de Sompel, H., Hagberg, A., Chute, R.: A principal component analysis of 39 scientific impact measures. *PloS one* **4**(6), e6022 (2009)
- [2] Canali, C., Casolari, S., Lancellotti, R.: A quantitative methodology to identify relevant users in social networks. In: *Business Applications of Social Network Analysis (BASNA)*, 2010 IEEE International Workshop on, pp. 1–8. IEEE (2010)
- [3] González, J.E.J., Santana, G.R.: *Spanish journal of psychology* (2002)
- [4] Gray, R.M., Davisson, L.D.: *An introduction to statistical signal processing*. Cambridge University Press (2004)
- [5] Gueye, I., Ndong, J., Sarr, I.: An accurate probabilistic model for community evolution analysis in social network. In: *2015 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pp. 343–349. IEEE (2015)
- [6] Jensen, D.R., Solomon, H.: A gaussian approximation to the distribution of a definite quadratic form. *Journal of the American Statistical Association* **67**(340), 898–902 (1972)
- [7] Kunisch, K., Volkwein, S.: Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical analysis* **40**(2), 492–515 (2002)
- [8] McGloin, J.M., Kirk, D.S.: An overview of social network analysis. *Journal of Criminal Justice Education* **21**(2), 169–181 (2010)
- [9] Ringberg, H., Soule, A., Rexford, J., Diot, C.: Sensitivity of pca for traffic anomaly detection. In: *ACM SIGMETRICS Performance Evaluation Review*, vol. 35, pp. 109–120. ACM (2007)
- [10] Sharma, S., Gupta, R.: Improved bsp clustering algorithm for social network analysis. *International journal of grid and Distributed Computing* **3**(3), 67–76 (2010)
- [11] Soule, A., Salamatian, K., Taft, N.: Combining filtering and statistical methods for anomaly detection. In: *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*, pp. 31–31. USENIX Association (2005)
- [12] Wasserman, S., Faust, K.: *Social network analysis: Methods and applications*, vol. 8. Cambridge university press (1994)

Vertex-centred Method to Detect Communities in Evolving Networks

Maël Canu, Marie-Jeanne Lesot and Adrien Revault d'Allonnes

Abstract Finding communities in evolving networks is a difficult task and raises issues different from the classic static detection case. We introduce an approach based on the recent vertex-centred paradigm. The proposed algorithm, named DynLOC-NeSs, detects communities by scanning and evaluating each vertex neighbourhood by means of a preference measure, using these preferences to handle community changes. We also introduce a new vertex neighbourhood preference measure, CWCN, more efficient than current existing ones in the considered context. Experimental results show the relevance of this measure and the ability of the proposed approach to detect classical community evolution patterns such as grow-shrink and merge-split.

1 Introduction

A main task in computational network analysis is community detection, that consists in identifying denser subnetworks related to a specific role (eg. common interests in social networks, groups of interacting proteins in biological networks...) Though there is no universal definition for community, many have been proposed: intuitively, a community is a group of entities whose members have more relations between them than with the rest of the network. Many definitions and methods exist and keep being proposed [3, 9].

Most community detection methods to date were designed to process static networks (see Section 2), however complex networks change over time and require methods able to take into account their dynamic (also referred to as temporal or evolutionary) dimension. It has been proved that straightforward use of static community

Maël CANU (e-mail: mael.canu@lip6.fr) · Marie-Jeanne Lesot (e-mail: marie-jeanne.lesot@lip6.fr)
Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France
CNRS, UMR 7606, LIP6, F-75005, Paris, France

Adrien Revault d'Allonnes (e-mail: allonnes@ai.univ-paris8.fr)
Université Paris 8, EA 4383, LIASD, F-93526, Saint-Denis, France

detection algorithms at each time step (re-computation) is not relevant, in particular the output partition is not stable [1].

In this paper, we propose two contributions: first, an event-based detection algorithm relying on a vertex-centred process allowing a fast computation and a decentralised implementation, as well as a preference measure, **Community-based Weighted Common Neighbours (CWCN)** used in the vertex-centred process and more efficient than existing measures in the considered context.

The rest of this paper is organised as follows. Section 2 presents related works about static and dynamic community detection methods. Section 3 describes the principles of the proposed method DynLOCNeSs, and introduces the vertex neighborhood measure CWCN. Experimental results to assess the ability of the method to capture simple network dynamics are provided in Section 4.

2 Related Works

We first present here static and dynamic community detection methods relevant to this paper. Other classic methods are reviewed in [3, 9, 27]. Then, we review an approach more related to the proposed method: the vertex-centred paradigm.

Static Paradigms Numerous static community detection approaches exist in the literature. They can be generic graph partitioning algorithms or take into account typical characteristics of the type of network they are designed for, such as power-law degree and small world effect in the case of social networks.

The main community detection method family is criterion optimisation. A global or local criterion measuring the quality of a graph partition into communities, such as the well-known modularity [9], is optimised through several iterations of an algorithm until convergence. Many existing criteria yield good quality partition (compared to a ground truth for example), but suffer from different drawbacks such as being subject to local extremum or resolution limit [10]. This kind of method is also known to be time-consuming [9].

More recently, label propagation methods [13, 24, 27] offer a decentralised alternative. They rely on propagation of a node identifier (“label”) from each vertex to every other in the network. However, despite being fast and suitable for detection in a decentralised environment, they have been found not to be stable as well [18, 25]. Moreover, they make massive use of propagation and can overflow the network with unnecessary traffic, especially in a decentralised environment.

Dynamic Paradigms The changeover from the static to the dynamic case is not easy. In particular, it depends on hypothesis about the graph evolution model. The most widespread one considers a dynamic graph as a collection of static graphs, discretising the dynamic aspect with one graph instance per time step. Naive static detection on each time step, named static re-computation, has quickly been found to be unstable [1], especially when using optimisation methods, because the identified community structure varies too much, unrelatedly to the community evolution. For example, a good modularity value can be achieved on several very different com-

munity partitions of the same graph. To address this issue, concepts like *temporal smoothness* introduced by Chakrabarti for evolutionary clustering were integrated [6].

But even more than in the static case, taking into account the nature of the considered networks and the dynamics they are subject to is essential to design efficient methods [19]. In this context, decentralised methods adapted to process the dynamic case have been found to offer good performance, in terms of partition quality as well as computational efficiency, also offering the advantage of being easily implemented in parallel frameworks, as it is the case for label propagation [7, 18]. It is also very popular for applications in specific environments such as small decentralised mobile networks, like Pocket Switched Networks (PSN), for which community detection helps to improve network discovery and information routing [15, 21].

Vertex-centred Methods Finally, vertex-centred approaches have gained popularity as a promising new community detection method family. They rely on the principle that some vertices in the network are “leaders” or “seeds” and the rest are followers [26]. Communities are formed by gathering followers around leaders, like in the *Top-Leaders* approach [23]. Although this method is more related to *k*-means clustering (re-allocating the leaders) than to a true leader-follower design, the introduced idea of expanding communities around leaders considering the potential *preference* of a follower vertex (resp. a group of follower vertices) to join a leader vertex has been exploited by numerous algorithms. *YASCA* [16] greedily expands communities around seeds and gather communities using ensemble clustering. *LICOD* [28] starts with a careful selection of leaders before computing ranked community membership for each follower, then adjusting preferences and memberships using strategies borrowed from social choice theories until stabilisation. *EMc* and *PGDc* [17] locally expand around seed via EM or Projected Gradient Descent algorithm, using conductance to delimit communities. Canu et al. [4] consider each vertex as a potential leader and build preference dependencies allowing to form communities. True leaders are the core of the dependencies, where the rest can be considered as followers.

Vertex-centred methods have also attracted attention to develop new dynamic community detection algorithms: for instance *Evo-Leaders*, an adaptation of Top-Leaders [11], *mux-LICOD*, an adaptation of LICOD for multiplex networks enabling use on evolving networks [14], *OLEM/OLTM* [22] that locally optimises modularity and the original approach of [29] based on weighted-edge graphs, using weight update rules to cope with the dynamicity together with a fitness function to ensure partition quality.

We can also cite agent-based approaches like *iLCD* that consider each vertex as an agent and apply dynamic evolution rules to simulate the community formation, yielding a community structure [5].

The major drawback of these algorithms is that they lose one of the initial benefits of the leader-based approach, i.e. lightness and flexibility. Built on top of Top-Leaders, *Evo-Leaders* [11] adds a costly split-merge of community at each time step. *mux-LICOD* [14] uses degree centrality and shortest path calculation to compare leaders and followers. Shortest path computation can be costly if used for each vertex to each

potential leader. It also relies on an aggregation phase repeated until stabilisation, though experiments do not reveal whether the stabilisation is fast or not. Finally, Zakrzewska et al.'s method [29] relies on a fitness function and a set of ad-hoc update rules and pruning over updates. It is hard to know however how efficient this policy is, as the experiments proposed by the authors are limited to a comparison with re-computation of the static counterpart. While faster than static re-computation, which is generally expected for specifically designed algorithms), the proposed F -score comparison with the set of static re-computed instances is not meaningful, as static re-computation has been proved to give unstable results [1].

3 Proposed Approach

This section describes the proposed approach, after defining the considered dynamical model. We sketch its principles and describe in details the algorithm, called DynLOCNeSs, which requires a vertex neighbourhood preference measure. We discuss such preference measures and introduce a new one, CWCN.

3.1 Principles

In the following, $G = (V, E)$ denotes an undirected graph, $\Gamma(v)$ for $v \in V$, the set of v 's neighbours and d_v , the degree of v . C denotes the set of detected communities and $C(v)$ the community of v . $S \subset V$ is the leader set, of all vertices being a leader for at least one other vertex. Each leader $s \in S$ has a set of followers $F(s) \subset V$. Alternatively, a follower f has a set of preferred leaders, denoted $L(f) \subset V$. Preference measures between two vertices are denoted using a function $\sigma : V \times V \rightarrow \mathbb{R}^+$.

Dynamicality We call *time step* $t_i, i \in \mathbb{N}$ a date corresponding to a given state of the graph G . The next time step t_{i+1} occurs when at least an edge changes (appears or disappears). The vertex events are treated as consequences of the edge moves: a vertex addition is captured as a new edge connecting a formerly isolated vertex. A vertex removal is captured in the same way, as the deletion of the last edge connecting this vertex to the rest of the graph. All edges are equally important, whether old or new. This model is widely used [12].

We denote $G_i = (V, E_i)$ the state of G and C_i the state of communities at time t_i , eg. G_0 is the initial graph at t_0 . Note that the time interval $|t_i - t_{i-1}|$ is not necessarily constant.

3.2 Proposed Algorithm: DynLOCNeSs

We propose DynLOCNeSs (**D**ynamic **L**ocation of **C**ommunities in **N**etwork **S**tructures), a vertex-centred approach to detect communities in dynamic graphs, more precisely a leader-based approach using a vertex neighbourhood preference measure.

The idea is to change from a batch to an event-based detection and modification process, and to perform the detection with as little as possible re-computation. Each vertex must determine whether it should change its leader. If so, it may also change community.

The proposed method takes as input an initial graph, G_0 , along with initial community structure C_0 and leader set S_0 , and only deals with the detection over time. These initial states can be computed using any leader-based method (see Section 2). The implementation presented here uses an approach in which each vertex $v \in V$ is considered as a potential leader and evaluates its neighbourhood, like *iLCD* [5] or Canu et al. [4]. It has the advantage of not pre-selecting a set of leaders, thus not suffering from the bad seed selection issue.

The main part of the algorithm is the *vertex update procedure* described in Algorithm 7). It is run when an edge (dis)appears, which is the only event considered here. The algorithm also relies on a times-step related vertex marking, which is used to identify whether the leaders or community must be re-computed. The marking is explained first, and then the vertex update procedure.

Marking A vertex is marked to signify it has changed community, and is meant to be seen only by the vertex neighbours. The marks made at t_i are visible at time t_{i+1} . Vertices having a marked vertex in their leader set will reconsider their community membership. This marking is the way to accelerate changes propagation through the graph, because a community change for a vertex increases the probability of one of its neighbours to change community too.

Vertex Update Procedure This key procedure is run for a vertex v , either leader or follower, only if a change occurred in its neighbourhood, the only possibility that may lead to a community change for v . In this case, at time t_i , each vertex v locally computes all the preferences between itself and its neighbours, ie. all the $\sigma(v, v')$ for all $v' \in \Gamma(v)$ (see Section 3.3 for discussion about σ). Because of the neighbourhood change, a leader (ie. a neighbour vertex maximising $\sigma(\cdot, v)$) may have disappear or a new one appear. If the new preference values imply a change in $L(v)$, the community of v is also re-evaluated. If that results in $C(v)$ changing, then v marks itself.

Flexibility and Local Computation. The proposed algorithm only uses local computations from each vertex, thus keeping the vertex-centred methods flexibility advantage. This allows an easy decentralised implementation in Pregel-like frameworks (see [20]): the vertex program is simple to write and few informations are susceptible to be shared between parallel processes.

3.3 Preference Measures

The proposed method relies on a vertex neighbourhood preference measure $\sigma : V \times V \rightarrow \mathbb{R}^+$, evaluating at which point a vertex $v \in V$ is close to a given neighbour $u \in \Gamma(v)$. It must reflect a closeness or attraction dynamics at work in the graph. For example, in a social network, $\sigma(v, u)$ must account for the friendship level of

Algorithm 7 Vertex Update Procedure for time step t_i **Require:**

- $v \in V$, a vertex
- $\Gamma_i(v)$, its neighbours at time t_i

Ensure:

- $C_i(v)$, updated community for v
- 1: **if** $\Gamma_i(v) \neq \Gamma_{i-1}(v)$ **then**
- 2: recompute v 's preferred leaders: $L(v) \leftarrow \arg \max_{u \in \Gamma_i(v)} \sigma(v, u)$
- 3: **if** $L(v)$ changes **or** any $u \in L(v)$ is marked **then**
- 4: $C_i(v) \leftarrow$ most frequent community among $L(v)$
- 5: **if** $C_i(v) \neq C_{i-1}(v)$ **then**
- 6: mark each v for time t_i
- 7: **end if**
- 8: **end if**
- 9: **end if**

v towards u . Such closeness often relies on the quantity of common neighbours between u and v as detailed below:

We review here three measures as presented in [8] (Section 2.2), and propose a new proposed measure **Community-based Weighted Common Neighbours (CWCN)**, taking into account known information community. Section 4 presents results of the algorithm implementing each of these measures. The mathematical expression is given for each measure for any $u, v \in V$.

Jaccard coefficient of neighbours is an adaptation of the well-known *Jaccard Index* for neighbour vertices in a graph, and compares the number of common neighbours to the total number of neighbours of both u and v . It is defined as follows:

$$\sigma_{Jac}(u, v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|} \quad (1)$$

Adamic-Adar is an adaptation of the eponymous measure used for web search and link prediction. It sums the number of common neighbours between u and v , using a logarithmic function that gives more importance to ‘‘rarer’’ features, here to less connected neighbours. It is defined as follows:

$$\sigma_{AA}(u, v) = \sum_{w \in \Gamma(u) \cap \Gamma(v)} \frac{1}{\log(|\Gamma(w)|)} \quad (2)$$

The *Preferential Attachment* measure is based on the eponymous concept popularised by Barabási and Albert [2]: the tendency of entities having many connections to attract more new connections than weakly connected ones. It multiplies the neighbourhood sizes of u and v , meaning that preference hugely depends on vertex degree. Using this measure results in large agglomerations of vertices around hubs. It is defined as:

$$\sigma_{PA}(u, v) = |\Gamma(u)| \times |\Gamma(v)| \quad (3)$$

The proposed CWCN measure *Community-based Weighted Common Neighbours* is a common neighbour measure weighted by the degree of the vertex being compared. While similar to the common neighbours $|\Gamma(u) \cap \Gamma(v)|$, the degree weighting scheme “attracts” a vertex much more toward high degree leaders and thus higher density areas in the graph, related to communities. This follows Barabási & Albert’s preferential attachment principle [2] but is less strong than the preferential attachment measure described above. It is defined as:

$$\sigma_{CWCN}(u, v) = |\Gamma(u) \cap \Gamma(v)| \times d_v \quad (4)$$

4 Experiments

This section presents several experiments supporting the validity of the proposed method. It compares the effectiveness of various preference measures presented Section 3.3. The goal of these experiments is to prove the ability of DynLOC-NeSs (together with an appropriate preference measure) to capture the dynamics of evolution of the network, and as such is done on small interpretable graphs, with experiments similar to [12]. The experiments on big graphs (data mining) are left to future works.

4.1 Protocol

Datasets. We use artificial benchmark graphs to assess the properties and validity of the proposed algorithm. They are obtained using the generator proposed by Granell et al. [12]. It keeps the vertex set constant and uses two community evolution patterns: grow-shrink, where some communities grow (gain vertices) while others shrink (lose vertices), and merge/split, where merge and splits occur between communities. It can generate an evolving graph of controlled size and density after one or both patterns, together with the ground truth community structure. We specify for each experiment the benchmark parameters used to generate the graphs.

Evaluation Criteria. We use the same criteria for partition comparison as in [12]: the classical information entropy-based measures *Normalised Variation of Information* (NVI) and *Normalised Mutual Information* (NMI), both bounded between $[0, 1]$. However, opposite to NVI, a NMI value of 1 indicates that the two partitions contain the same information (identical) whereas 0 indicates that the partitions are totally dissimilar. A good community structure partition thus minimises NVI and maximises NMI. For the mathematical expressions, see [12].

We choose not to use the proposed windowed variant [12] as it does not bring significant benefit and it is difficult to interpret. As a matter of fact it requires to carefully select the time window value, which plays a significant role in the performance evaluation.

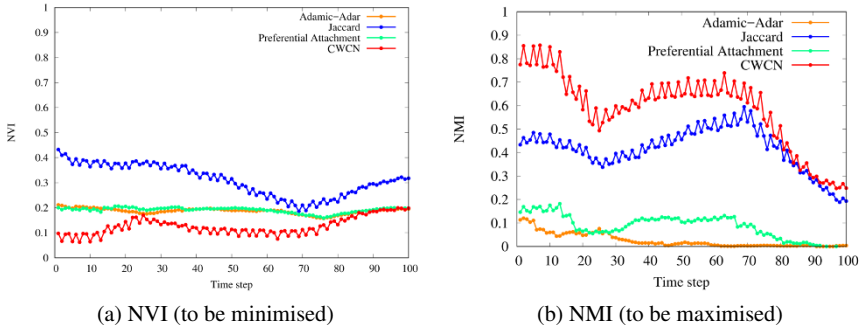


Fig. 1: Comparison for the grow-shrink pattern on 100 time steps

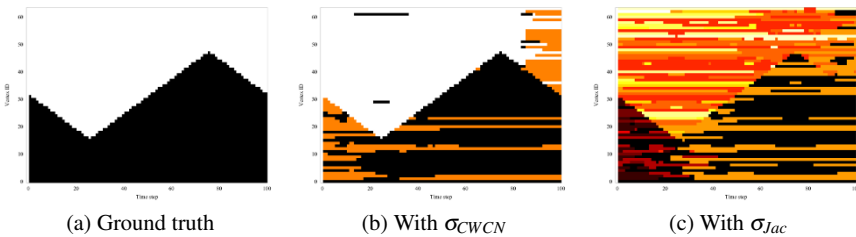


Fig. 2: (Colors online) Comparative visualisation of the community repartition between σ_{Jac} and σ_{CWCN} evaluated on the grow-shrink process.

4.2 Preference Measure Comparison

The first experiment is performed in order to compare the effect of the different preference measures exposed in Section 3.3. We use here the classic planted bisection model [7, 12]. In this model, the graph is divided into two communities and the algorithm has to correctly classify each vertex as belonging to one or the other.

The proposed algorithm is tested for each preference measure on two evolution patterns : grow-shrink and merge-split. For each pattern, 10 instances of a graph of 64 vertices are generated, with intra-community density of 0.5 and inter-community density of 0.05, for 100 time steps. These values are the ones used in [12]. The ground truth, shown on Fig. 2a and 4a, is thus made of 2 communities of 32 vertices each at t_0 .

Results for the Grow-Shrink pattern are presented on Figures 1 and 2 (the measures not included in Figure 4 produce only one community at each time step, therefore the colormap is all black) are the mean of NVI and NMI runs over the 10 graphs, and a colormap visualisation where each pixel color represents the community

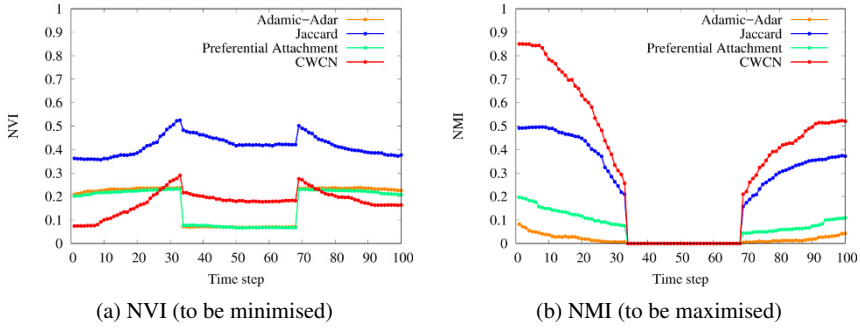


Fig. 3: Comparison for the merge-split pattern over 100 time steps

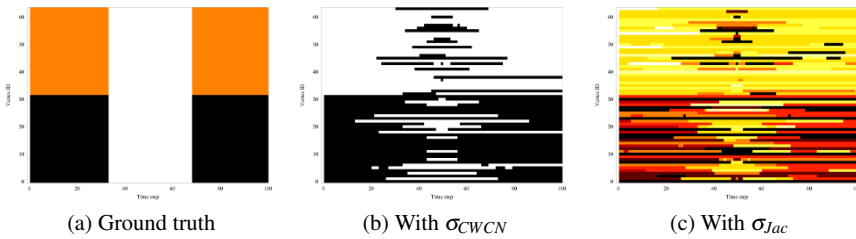


Fig. 4: (Colors online) Comparative visualisation of the community repartition between σ_{Jac} and σ_{CWCN} evaluated on the merge-split process.

assignment of a vertex (id on the y axis) at a given time step (on the x axis). We can see that DynLOCNeSs with σ_{CWCN} globally detects the grow-shrink bisection pattern, except that a third community (orange) is identified. This community in fact replaces the black one at the beginning and the white one at the end: the method takes the grow-shrink evolution as a transfer between two communities via a third one, impacting NVI and NMI values. However, the clearly visible grow-shrink triangle shapes indicate that the evolution pattern has correctly been identified. This is less obvious for the method with σ_{Jac} . It detects 14 communities and even if the triangle shape can be guessed there is a lot of noise and community misassignment.

The other two cases, σ_{AA} and σ_{PA} , are not pictured because they assign every vertex to a single community, resulting in an entire black colormap.

The merge-split process is presented in Figures 3 (criteria) and 4 (visualisation). Again, the measures not included in Figure 4 produce only one community at each time step, therefore their colormap is black.

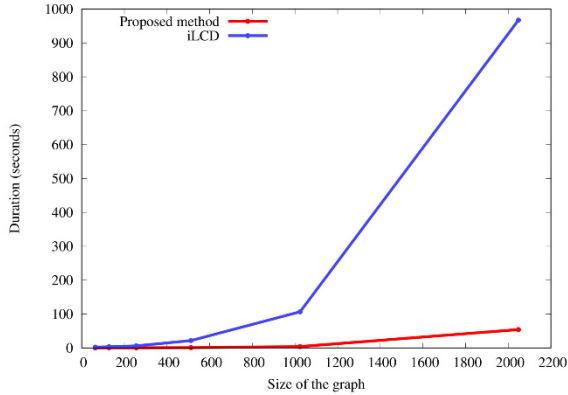


Fig. 5: Speed of execution as a function of the graph size

Merge-Split is less successfully detected than grow-shrink. We observe that σ_{CWCN} finds two communities where σ_{Jac} finds ten, but the abrupt merge is not correctly identified, whereas the CWCN variant yields less noise than Jaccard one.

Let aside the merge, the CWCN variant nonetheless achieves better NMI and NVI than the other methods. The perfect NVI for σ_{AA} and σ_{PA} during the merge (fig. 3 (b)) can be explained by the fact that both methods only detect one community at any time step. This is prejudiced when two communities exist, but it is correct during the merge. It is a side effect related to the chosen planted bisection, but inherently denotes a poor quality of detection for these two criteria.

Execution Time Because the input and output of dynamic community detection algorithms depend on the dynamicity model used, it is difficult to compare them. For example, *iLCD* input is event-based (edge addition or deletion) and its output is a chronological sequence of community states. A state change can happen any time an edge is removed. The consequence is that, if launched on a time step sequence similar to those used to test the proposed algorithm, the community structure can vary several times during a same time step. Any heuristic to gather all the changes made during a time step would inevitably erase information and introduce a bias.

Another example, the multi-step adaptation of Louvain algorithm [1] takes a sequence of time steps into account, but outputs a unique community structure at the end of the process and it is not possible to track the evolution of this structure during the detection process.

A more neutral comparison axis is the execution time, presented below, chosen to illustrate the performance of the proposed method : six graphs, of 64, 128, 256, 512, 1024 and 2048 vertices respectively, were generated with the same density as in the previous experiments: 0.05 intra-community and 0.5 inter-community (e.g. 375,000 edges for 1,024 vertices), over 10 time steps.

We measure the mean time, over 5 runs, taken by DynLOCNeSs and by iLCD to process each graph. The platform used is a Intel Core i7-2600K CPU @ 3.40GHz Workstation with 16GB RAM.

Results are presented on Figure 5. We can see that iLCD processing time is skyrocketing before the method we propose, which is a significant advantage to process either large graphs or large number of time steps.

5 Conclusion and Future Works

We propose a new dynamic community detection method, named DynLOCNeSs that consists in a vertex-centred approach to re-compute only a small local fraction of vertex neighbourhood. The algorithm relies on a vertex neighbourhood preference measure. We introduced a novel one, CWCN. Experiments on benchmark graphs show that CWCN yields better results than the other measures and that the overall method is well able to detect common patterns in community evolution such as grow-shrink and merge-split.

We are considering additional work on the community evolution patterns to better capture the dynamics and improve the quality of DynLOCNeSs pattern identification. We are also working on experiments to assess the performance of the method on large graphs, up to several millions of vertices. We also plan to evaluate the CWCN criterion over other clustering-related problems.

Acknowledgements This work was performed as part of the Homo Textilus project, supported by the French ANR agency under the grant ANR-11-SOIN-007.

References

- [1] Aynaud, T., Guillaume, J.L.: Static community detection algorithms for evolving networks. In: Proc. of the 8th Intl. WiOpt'10 Symposium, pp. 513–519 (2010)
- [2] Barabási, A.L., Albert, R.: Emergence of Scaling in Random Networks. *Science* **286**(5439), 509–512 (1999)
- [3] Bedi, P., Sharma, C.: Community detection in social networks. *WIREs Data Mining Knowl. Discov.* **6**(3), 115–135 (2016)
- [4] Canu, M., Detyniecki, M., Lesot, M.J., Revault d'Allonnes, A.: Fast community structure local uncovering by independent vertex-centred process. In: Proc. IEEE/ACM Intl. Conf. ASONAM'15, 823–830. ACM (2015)
- [5] Cazabet, R., Amblard, F.: Simulate to Detect: A Multi-agent System for Community Detection. In: Proc. IEEE/WIC/ACM WI-IAT'11, vol. 2, 402–408 (2011)
- [6] Chakrabarti, D., Kumar, R., Tomkins, A.: Evolutionary Clustering. In: Proc. 12th ACM SIGKDD Intl. Conf. KDD '06, 554–560. ACM (2006)
- [7] Clementi, A., Di Ianni, M., Gambosi, G., Natale, E., Silvestri, R.: Distributed community detection in dynamic graphs. *Theoretical Computer Science* (2014)
- [8] Cohen, S., Kimelfeld, B., Koutrika, G.: A Survey on Proximity Measures for Social Networks. In: Search Computing, no. 7538 in LNCS, 191–206. Springer Berlin Heidelberg (2012)
- [9] Fortunato, S.: Community detection in graphs. *Phys. Rep.* 75–174 (2009)
- [10] Fortunato, S., Barthélemy, M.: Resolution limit in community detection. In: Proc. Natl. Acad. Sci. **104**(1), 36–41 (2007)
- [11] Gao, W., Luo, W., Bu, C.: Evolutionary community discovery in dynamic networks based on leader nodes. In: Proc. 2016 Intl. Conf. BigComp, 53–60 (2016)

- [12] Granell, C., Darst, R.K., Arenas, A., Fortunato, S., Gómez, S.: Benchmark model to assess community structure in evolving networks. *Phys. Rev. E* **92**(1), 012,805 (2015)
- [13] Gregory, S.: Finding overlapping communities in networks by label propagation. *New J. Phys.* **12**, 1–26 (2010)
- [14] Hmimida, M., Kanawati, R.: Community detection in multiplex networks: A seed-centric approach. *Networks and Heterogeneous Media* **10**(1), 71–85. AIMS (2015)
- [15] Hui, P., Yoneki, E., Chan, S.Y., Crowcroft, J.: Distributed Community Detection in Delay Tolerant Networks. In: *Proc. ACM/IEEE Intl. Workshop MobiArch'07*, p. 7. ACM (2007)
- [16] Kanawati, R.: YASCA: an ensemble-based approach for community detection in complex networks. In: *Computing and Combinatorics*, 657–666. Springer (2014)
- [17] van Laarhoven, T., Marchiori, E.: Local community detection by seed expansion: from conductance to weighted kernel 1-mean optimization. (submitted, ArXiv: 1601.05775) (2016).
- [18] Leung, I.X.Y., Hui, P., Liò, P., Crowcroft, J.: Towards real-time community detection in large networks. *Phys. Rev. E* **79**(6), 066,107 (2009)
- [19] Lin, Y.R., Chi, Y., Zhu, S., Sundaram, H., Tseng, B.L.: Facetnet: a framework for analyzing communities and their evolutions in dynamic networks. In: *Proc. 17th Intl. Conf. WWW'08*, 685–694. ACM (2008)
- [20] McCune, R.R., Weninger, T., Madey, G.: Thinking Like a Vertex: A Survey of Vertex-Centric Frameworks for Large-Scale Distributed Graph Processing. *ACM Comput. Surv.* **48**(2), 25:1–25:39 (2015)
- [21] Orlinski, M., Filer, N.: The rise and fall of spatio-temporal clusters in mobile ad hoc networks. *Ad Hoc Networks* **11**(5), 1641–1654 (2013)
- [22] Pan, G., Zhang, W., Wu, Z., Li, S.: Online Community Detection for Large Complex Networks. *PLoS ONE* **9**(7), e102,799 (2014)
- [23] Rabbany, R., Chen, J., Zaïane, O.R.: Top leaders community detection approach in information networks. In: *Proc. 4th SNA-KDD'10 Workshop* (2010)
- [24] Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E* **76**(3), 036,106 (2007)
- [25] Rezaei, A., Far, S.M., Soleymani, M.: Near Linear-Time Community Detection in Networks with Hardly Detectable Community Structure. In: *Proc. IEEE/ACM Intl. Conf. ASONAM'15*, 65–72. ACM (2015)
- [26] Riedy, J., Bader, D.A., Jiang, K., Pande, P., Sharma, R.: Detecting communities from given seeds in social networks. Technical Report, Georgia Institute of Technology (2011)
- [27] Xie, J., Kelley, S., Szymanski, B.K.: Overlapping Community Detection in Networks: The State-of-the-art and Comparative Study. *ACM Comput. Surv.* **45**(4), 43 (2013)
- [28] Yakoubi, Z., Kanawati, R.: LICOD: A Leader-driven algorithm for community detection in complex networks. *Vietnam J. Comput. Sci.* **1**(4), 241–256 (2014)
- [29] Zakrzewska, A., Bader, D.A.: A Dynamic Algorithm for Local Community Detection in Graphs. In: *Proc. IEEE/ACM Intl. Conf. ASONAM'15*, 65–72. ACM (2015)

Clustering, Prominence and Social Network Analysis on Incomplete Networks

Kshiteesh Hegde, Malik Magdon-Ismail, Boleslaw Szymanski and Konstantin Kuzmin

Abstract Social networks are a source of large scale graphs. We study how social network algorithms behave on sparsified versions of such networks with two motivations in mind:

1. In practice, it is challenging to collect, store and process the entire often constantly growing network, so it is important to understand how algorithms behave on incomplete views of a network.
2. Even if one has the full network, algorithms may be infeasible at such large scale, and the only option may be to sparsify the networks to make them computationally tractable while still maintaining the fidelity of the social network algorithms.

We present a variety of methods for sparsifying a network based on linear regression and linear algebraic sampling for graph reconstruction. We *compare the methods against one another* with respect to clustering. Specifically, given a graph G , we sample the columns of its adjacency matrix and reconstruct the remaining columns using only those sampled columns to obtain \hat{G} , the reconstructed approximation of G . We then perform clustering on G and \hat{G} to get two sets of clusters and compute their modularity, fitness and centrality. Our thorough experimentation reveals that graphs reconstructed through our methodology preserve (in some cases, even improve) community structure while being orders of magnitude more efficient both in storage and computation. We show similar results if the target is prominence of nodes rather than clusters.

Kshiteesh Hegde (e-mail: hegdek2@rpi.edu)✉ · Malik Magdon-Ismail (e-mail: magdon@rpi.edu) · Boleslaw Szymanski (e-mail: szymab@rpi.edu) · Konstantin Kuzmin (e-mail: kuzmik@rpi.edu)

Dept. of Computer Science, Rensselaer Polytechnic Institute, Troy, NY

1 Introduction

The ever increasing popularity of social networks has resulted in increasing availability of massive graphs. Their sheer size renders them unwieldy for carrying out downstream machine learning operations. Further, such networks are difficult to measure entirely and often we can only access partial snapshots. We need ways to extract information from partially observed networks. This is one of the key motivations for our work, which is to address the question: Are there ways of sampling the edges of the network (perhaps re-weighting them) so that machine learning on the sparsified (incomplete) network produces results that are faithful to the full network.

The task of sampling a graph has applications across many domains. For example, in a social network with a billion nodes, questions arise like: who are a person’s potential friends or who are the leaders and influencers of a given group of people? In a very large research collaboration network, we may want to know which researchers are leaders in a particular field or who are the best collaborators between different fields. In a product rating setting, sellers may want to know which products (movies, books) in one genre are a gateway to another genre. Getting a bird’s eye view of these large networks (and many other types of networks) can be instrumental in solving interesting problems quickly. In this paper, we propose ways to address these problems using techniques from graph sparsification and reconstruction.

As discussed later in the related work section, there is a body of research accumulating in the Linear Algebra community which tries to approach this problem by treating the networks as matrices. The benefit is that the spectral structure of matrices can be preserved up to a finite rank if the samples are chosen carefully. In this work, we use these techniques in the social network analysis (SNA) setting. We also use linear regression in one of our methods where we choose a subset of the columns of the adjacency matrix of the full dataset and regress on the remaining columns to get our estimate.

The essence of our work is visualized in Fig. 1. The dataset could be in multiple formats but we represent it as an adjacency matrix A with $A(i, j) = A(j, i) = w$ if there is an edge e between i^{th} and j^{th} node with weight $w > 0$. Let r_i be the i^{th} row of A . We choose a small subset of the columns of A . This corresponds to choosing certain nodes from the graph and all the edges that those nodes are involved in. Then using the symmetric property of the adjacency matrix, linear regression and linear algebraic sampling methods (see section 3) we reconstruct the missing edges and nodes. The weights of the edges in the new graph will change depending on the probabilities with which the rows of A are chosen. We call the adjacency matrix of our reconstructed graph (which corresponds to the modified dataset) \hat{A} . Now, to evaluate the performance of our method, we compute clustering metrics on this new dataset. We compare them with those obtained from the full dataset.

We formulate two problems in this study:

1. Given A , sparsify to \hat{A} , so that machine learning tasks on \hat{A} are faster and produce almost as accurate predictions as from A .

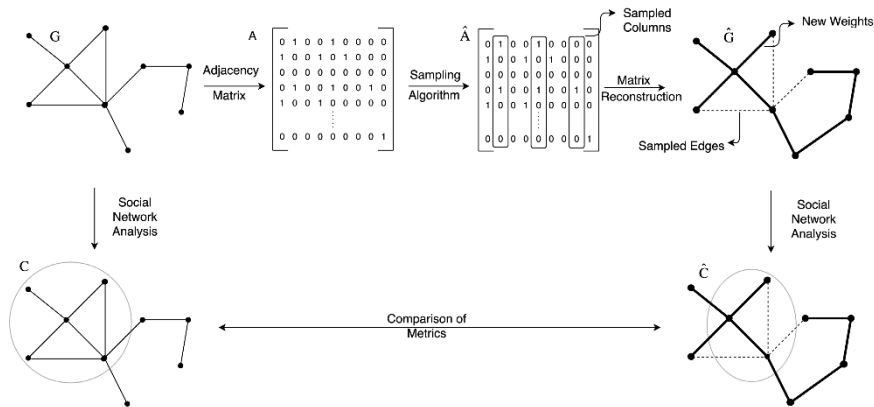


Fig. 1: Outline of our workflow. The dataset is represented as an adjacency matrix. The columns of this matrix are sampled to yield a new adjacency matrix which has new weights for its edges. A new graph is constructed by reconstructing the missing edges using this adjacency matrix. The clustering metrics are computed on both the full and sampled graphs.

2. Knowing nothing about A , identify a few columns to sample to get \hat{A} , so that machine learning tasks on \hat{A} are more efficient and produce almost as accurate predictions as from A .

To give a better idea of this process, we illustrate it by using a toy graph accompanied by the corresponding adjacency matrix A in Fig. 2. Each of the edges in the graph is assumed to have a unit weight unless it is shown thicker in which case it would mean that it has a weight $w > 1$.

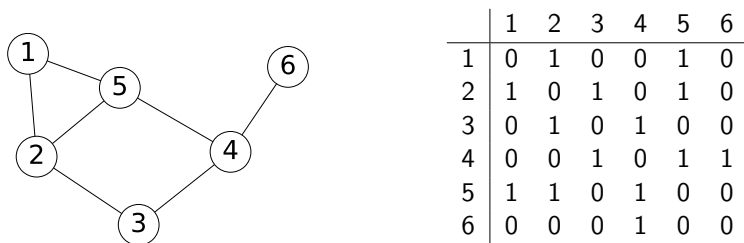


Fig. 2: A toy graph and its associated adjacency matrix

Let the columns 1,2,3 and 6 be sampled from A . The entries in the unseen columns 4 and 5 are partially populated by using the symmetric nature of A . We apply our reconstruction techniques on these sampled columns and the partially populated columns to get estimates of A . In one method (Algorithm 8), we use linear regression to guess the unseen edges and in the other method (Algorithm 9) we rescale the weight of the seen edges by $\sqrt{1/p_i}$ where p_i is the probability of i^{th} being chosen to “make up” for the lost edges. For example, if columns from A are chosen uniformly,

then $p_i = 1/6$. Let the estimates obtained from these two very different approaches be \hat{A}_1 and \hat{A}_2 respectively. We show the corresponding graphs in Fig. 3. Finally, we perform clustering (see Section 3.4) on \hat{A}_1 and \hat{A}_2 and compute some metrics (see Section 3.5) to measure the performance of our algorithms. Fig. 3 also shows the clustering on \hat{A}_1 and \hat{A}_2 . Note that all the nodes of the same color belong to the same cluster.

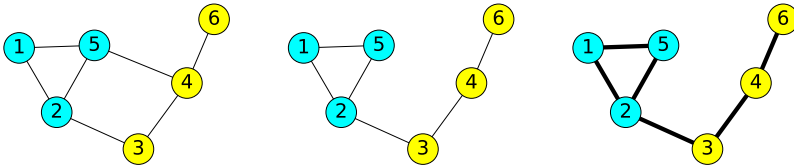


Fig. 3: Toy graph and clustering for its estimates

Our Contribution and Summary of Results

In this work we examine the feasibility of sampling and reconstructing large graphs when we do not have access to the entire graph while proposing two methods to address the problem. We simulate the issue of having incomplete graphs by choosing a subset of the full graph and working only with this small subset to build the unseen graph. Specifically, we choose some well-known metrics pertaining to graphs and use them as a yardstick to measure the performance of the different sampling algorithms which treat graphs as matrices. Our main contribution is to show that it is feasible to extract useful information from incomplete graphs and designing two algorithms to do so.

Some of the key observations that we were able to make are discussed below. We were able to improve the modularity of the clusters even when progressively sampling only 0.15% of the nodes and their related edges. The expansion of the clusters actually improved and was better in the sampled datasets. We were able to achieve this by using only a tiny fraction of the time required for processing the full graph. For example, the Amazon dataset (see section 3.6), which has over 300,000 nodes and 900,000 edges, took almost an hour to be evaluated while with just 0.45% of the data, we were able to evaluate it with reasonable accuracy in about 6 minutes. Some metrics were more robust to sparsification than others. Prominence (centrality) measures weren't preserved as well as clustering metrics. We believe the reason for this is that clustering is inherently more robust compared to centrality in the sense that it is less specific. A more detailed analysis can be found in section 4.

2 Related Work

There is some work done related to sampling of graphs. A few researchers [23], in a collaborative effort, compared breadth first search random walk based sampling

methods and their conclusions were not promising. In a relatively older work [20], the authors came up with a scheme where a few “landmark” nodes are selected beforehand and the shortest path distances between two nodes are estimated based on that at runtime. The “landmark” node in a way summarizes a few nodes and thus can be treated as a representative for those nodes. This is not sampling of edges or nodes per se but we are mentioning this work because it tries to make the graph “small” before going ahead with downstream computations. However, another work [21] actually samples the edges and keeps the number of nodes unchanged in order to achieve faster graph clustering. They rank the edges using a similarity heuristic and then retain a set number of edges per node. Another interesting work [22] treats the graph as an electrical network and computes effective resistances of the edges and sparsifies the graph. There has been some prior work [13] which looks at sampling of graphs but they only sample randomly and do not consider graphs as matrices. Another work [14] contains a comparison of community detection algorithms on graphs but does not take into consideration the issues arising from working with large scale graphs.

There is another line of work which looks at computing centrality measures on large graphs quickly and efficiently. For example, in [24] the authors try to use virtual nodes in graphs in an attempt to quickly compute betweenness centrality. They assume the graphs are large, sparse and lightly weighted and inject virtual nodes into them and then compute betweenness centrality. One of the breakthrough works [6] significantly reduces the time required to compute betweenness centrality. Later work [4] proposed further improvements, so we think running those algorithms on sampled graphs would greatly increase the size of datasets on which such computations are feasible; especially when combined with parallel methods like multi-threading [15].

Another approach, perhaps very relevant to the kind of sampling algorithms that we study in this paper, is the use of matrix sparsification techniques with a goal of sparsifying them as discussed in [1] and [3]. Finally, [16] covers a lot of randomized algorithms aimed at obtaining an approximation of a matrix.

We do not perform a thorough survey of all the clustering algorithms available as it is beyond the scope of this work. Interested readers can refer to [9] for such an analysis. In our work, we compare clustering metrics, as we will discuss soon, computed on large graphs and their reconstructed counterparts.

3 Methodology

In this section we discuss the algorithms, metrics, datasets and experimental setup used in this paper. We investigate two methods of solving the problem of reconstructing incomplete graphs. We use MATLAB[®] notation in Algorithms 8, 9.

1. **Linear Regression:** Given the square symmetric adjacency matrix $A \in \mathbb{R}^{n \times n}$ of graph G with n nodes, we randomly choose $k < n$ ($k \ll n$ if n is very large) columns of A . Let this be $X \in \mathbb{R}^{n \times k}$. We can use the symmetric property of A to partially fill out $Y \in \mathbb{R}^{n \times (n-k)}$. The indices of the k columns that were selected are stored. Now, we use linear regression to get an estimate \hat{Y} of Y . We have $\hat{Y} = X(X^\dagger Y)$ where X^\dagger represents the Moore-Penrose pseudo-inverse of X .

We get the estimate $\hat{A} \in \mathbb{R}^{n \times n}$ of A by using X , \hat{Y} and indices of k sampled columns. The above process is shown in Algorithm 8.

Algorithm 8 Linear Regression

```

1:  $A = \text{get\_adjacency\_matrix}(G)$  ▷ Graph  $G$  is given
2:  $K = \text{randsample}(N, k)$  ▷ Store the indices of  $k$  chosen columns
3:  $X = A(:, K); Y = \mathbf{0}^{k \times n-k}$  ▷ Get the  $k$  columns from  $A$ 
4:  $Y(K, :) = X(N - K, :)^T$  ▷ Use symmetric properties of  $A$  to partially fill  $Y$ 
5:  $\hat{Y} = X(X^\dagger Y)$  ▷ Perform Linear Regression to build unseen graph
6:  $\hat{A} = \mathbf{0}^{n \times n}$  ▷ The new adjacency matrix
7:  $\hat{A}(:, K) = X$  ▷ Sampled columns
8:  $\hat{A}(:, N - K) = \hat{Y}$  ▷ Reconstructed columns

```

Note that Algorithm 8 can be applied multiple times to the full adjacency matrix to get multiple reconstructions of the graph. These estimates can then be combined to get a new estimate. In fact, in section 4, we test this approach by taking up to three estimates while evaluating the performance.

2. **Linear Algebraic Sampling Method:** In this approach, we initially choose k columns randomly from A . Instead of working with two matrices X and Y like in 1, we work with only one $X_i \in \mathbb{R}^{n \times n}$ matrix. The way X_1 is built is as follows. The chosen columns are rescaled by a factor of the probability with which they were chosen. This acts as the reconstruction step because in a way we are accounting for the missing information by giving more importance to the entries that we have. In addition, since A is symmetric, we further fill X_1 using this information. Now, we use one of the sampling algorithms which will be described in Section 3.3 to get a set of probabilities to further sample A . Using this set of probabilities, we will have k more columns. We can build X_2 in a similar fashion to X_1 . With these two estimates of A , we can now build \hat{A} as follows.

$$\hat{A} = \alpha X_1 + (1 - \alpha) X_2 \tag{1}$$

where α can be varied between 0 and 1 to get a weighted average of the estimates. Note that this process can be repeated to get different estimates. This is shown in Algorithm 9.

3. **Sampling Algorithms:** The following sampling methods can be used in step 7 of Algorithm 9.
 - a. **Leverage Score Sampling (LVG):** Given an $m \times n$ matrix A with $m > n$, let U denote an $m \times n$ matrix consisting of the left singular vectors of A . If the row vector $U_{(i)}$ is the i^{th} row of the matrix U , then $l_i = \|U_{(i)}\|_2^2$ for $i \in \{1, \dots, m\}$ are the leverage scores [17] of the rows of A . The leverage scores signify the ‘‘influential’’ rows that can be ‘‘good representatives’’ of a matrix. We compute these scores for the given matrix and use them as probabilities for selecting a particular column from that matrix.

Algorithm 9 Linear Algebraic Sampling (LAS)

```

1:  $A = \text{get\_adjacency\_matrix}(G)$  ▷ Graph  $G$  is given
2:  $K = \text{randsample}(N, k)$  ▷ Store the indices of  $k$  chosen columns
3:  $X_1 = \mathbf{0}^{n \times n}$ 
4:  $X_1(:, K) = A(:, K)$  ▷ Get the  $k$  columns from  $A$ 
5:  $X_1(K, :) = A(K, :)^T$  ▷ Use symmetric properties of  $A$ 
6:  $X_1 = \text{diag}(P_1) \times X_1$  ▷  $P_1$  is the vector of rescaling factors of length  $n$ 
7:  $P_2 = \text{smp1\_algo}(X_1)$  ▷ Get a new set of probabilities using one of the sampling algorithms
8:  $K = \text{sample}(P_2, N - k, k)$  ▷ Get new unseen  $k$  columns w.r.t.  $P_2$ 
9:  $X_2 = \text{construct\_X}(K_2, P_2)$  ▷ Repeat steps 3 – 6 on  $X_2$ 
10:  $\hat{A} = \alpha X_1 + (1 - \alpha) X_2$  ▷ Reconstructed  $\hat{A}$ 

```

- b. **Dual-Set Sparsification (DSS)**: Described in [5], DSS is a deterministic algorithm that selects rows from matrices with orthonormal columns. It is based on [22] that we reviewed in Section 2. We recommend referring to Algorithm 1 in [5] to get more details about this method. In short, it returns a set of n weights out of which r are non-zero, which are the sampling probabilities for our purposes, for an $l \times n$ matrix A of rank k in $O(rnk^2 + nl)$ time.
- c. **Adaptive Sampling (AS)**: For a detailed discussion of this method refer to Section 2 in [8]. To summarize, this algorithm does sampling in multiple iterations and in an adaptive manner. The rows in each new iteration get picked with probabilities proportional to their squared distances from the span of the rows that have already been picked previously.

All the algorithms above come with some form of theoretical guarantees for preserving the spectral structure of the Laplacian [17], [5], [8].

4. **SpeakEasy**: This [10] is a label propagation clustering algorithm which robustly detects both overlapping and non-overlapping clusters. The nodes in SpeakEasy update their labels based on their neighbors' labels and take into account their global popularity in the network. Note that we do not aim to improve clustering performances, but use this state-of-the-art "off the shelf" method. It could be an interesting extension to this work to use different clustering algorithms.
5. **Performance Metrics**: To compare the quality of the sparsified graph \hat{G} with the ground truth G we use the clusters and prominence measures obtained from both graphs. Let the community partition be given for a network $G = (V, E)$ with $|E|$ edges. Let C be the set of all communities, c a specific community in C with $|c|$ number of nodes, $|E_c^{in}|$ the number of edges between nodes within community c , $|E_c^{out}|$ the number of edges from the nodes in community c_i to the nodes outside c .

- a. **Modularity (Q)** [18], [19]: Modularity for unweighted and undirected networks is defined as the ratio of difference between the actual and expected (in a randomized graph with the same number of nodes and the same degree sequence) number of edges within the community.

$$Q = \sum_{c \in C} \frac{|E_c^{in}|}{|E|} - \left(\frac{2|E_c^{in}| + |E_c^{out}|}{2|E|} \right)^2 \quad (2)$$

- b. **Contraction** [7]: It measures the average number of edges per node inside a community. The larger the value of this metric, the higher the quality of the community. For undirected networks (the ones examined in this work), this would be $\frac{2|E_c^{in}|}{|c|}$
- c. **Expansion** [7]: It measures the average number of edges outside a community. The smaller the value of this metric, the higher the quality of the community. Using the previous notation, expansion would be $\frac{|E_c^{out}|}{|c|}$
- d. **Conductance** [7]: It measures the fraction of the total number of edges that have an endpoint outside a community. A smaller value of conductance means a better community. Conductance is defined as $\frac{|E_c^{out}|}{2|E_c^{in}| + |E_c^{out}|}$
- e. **Intra-Density** [7]: The internal density of a community. The larger the value of this metric, the higher the quality of communities. For a particular community c , intra-density is defined as $\frac{2|E_c^{in}|}{|c|(|c|-1)}$
- f. **Fitness** [7]: The ratio between the internal degree and the total degree of a community. Higher the value of fitness, better the quality of the community. Fitness is defined as $\sum_{c \in C} \frac{|E_c^{in}|}{|E_c^{in}| + 2|E_c^{out}|}$

6. **Datasets:** We used a variety of data sets in our experiments ranging from e-commerce to collaboration networks to social networks. We summarize the datasets here.

- a. **Amazon** [12]: This is a product co-purchase network of amazon.com. If a product is frequently co-purchased with another product then those two products have an undirected edge between them. There are 334,863 nodes and 925,872 edges.
- b. **DBLP Collaboration Network** [25]: In this co-authorship network, two authors are connected if they have published at least one paper together. It has 317,080 nodes and 1,049,866 edges.
- c. **Political Blogs** [2]: This is a directed network of hyperlinks between weblogs on US Politics during 2004 general election. It has 1,224 nodes and 19,022 edges.
- d. **College Football** [11]: This network represents the schedule of games between college football teams in a single season. There are 115 nodes and 613 edges.
- e. **Zachary's Karate Club** [26]: This network represents the friendships between 34 members of a karate club at a US university during two years. It has 34 nodes and 78 edges.

4 Performance Analysis

In this section, we describe the experimental setup and the choices that were made for the experiments. We sampled between 0.15% and 30% of columns from the datasets. We chose 3 different k 's for each dataset: 500, 100 and ,2000 for Amazon and DBLP, 150, 225 and 300 for Political Blogs, 30, 40 and 50 for Football and 5, 7 and 10 for Karate. Also, for each dataset and each k , we ran 3 iterations of Algorithm 8. This way, we had 3 estimates of the dataset for each k . We also timed each process and the comparison between full datasets and their estimates is shown in Fig 5. In case of Algorithm 9, instead of running the same algorithm three times, we ran it only once for each of the sampling methods described before. Thus, we again obtained three estimates. Similar to the earlier process, we timed Algorithm 9 as well and the performance is shown in Fig 4. The parameter mentioned in Equation 1 was set to 0.3 to give importance to the latest reconstruction of the dataset. The dark bars represent the full datasets while the gray bars represent the best performing partial datasets. Y-axes in both Fig. 4 and Fig. 5 represent the value of the metrics.

After we had the estimates from either algorithm, we performed the task of clustering on them. We ran the clustering algorithm mentioned in Section 3.4 on new adjacency matrices to obtain new sets of clustering. Now, with the clustering set of the full graph and that from the estimated graph, we were able to compute the community quality measures defined in Section 3.5.

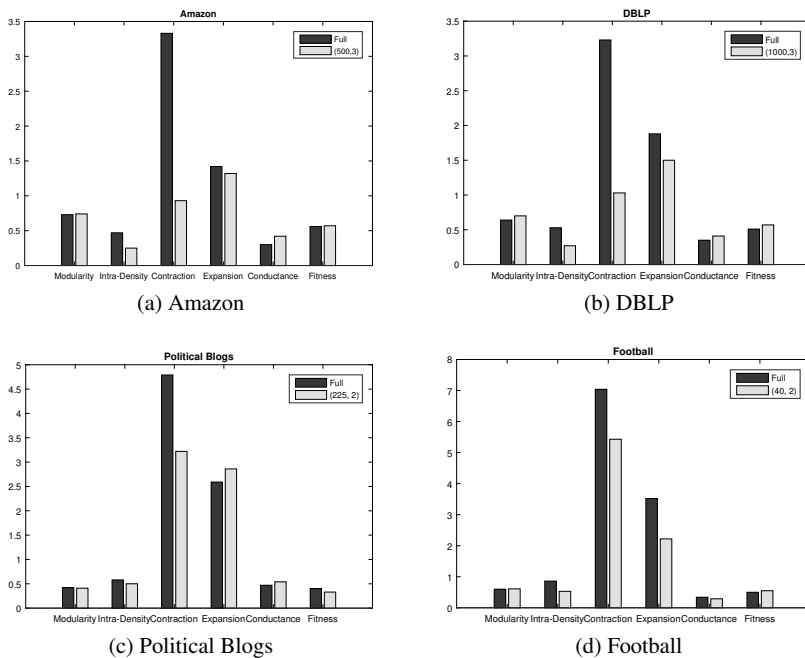


Fig. 4: Performance of Linear Algebraic Sampling Methods

1. **Clustering:** We can see that modularity, intra-density, expansion, conductance and fitness are all very well preserved irrespective of the algorithm or the dataset. We show the results for the best k and best performing sampling algorithm. We would like to note that the metrics are also preserved for other values of k . Readers can refer to the legend in each of the figure to see what k and how many iterations of running the algorithm (in case of Fig. 5) and with which sampling algorithm (in case of Fig. 4) produced the best results. For Fig. 4, we use the notation: 1=LVG, 2=DSS, 3=AS. This, combined with the fact that expansion has improved (lower the better) in almost every case shows that the reconstructed graphs have a better community structure. In case of algorithm 8 (Fig. 5), we learned that running at least 2 iterations provides the best results. We omit the results for the Karate dataset to conserve space.

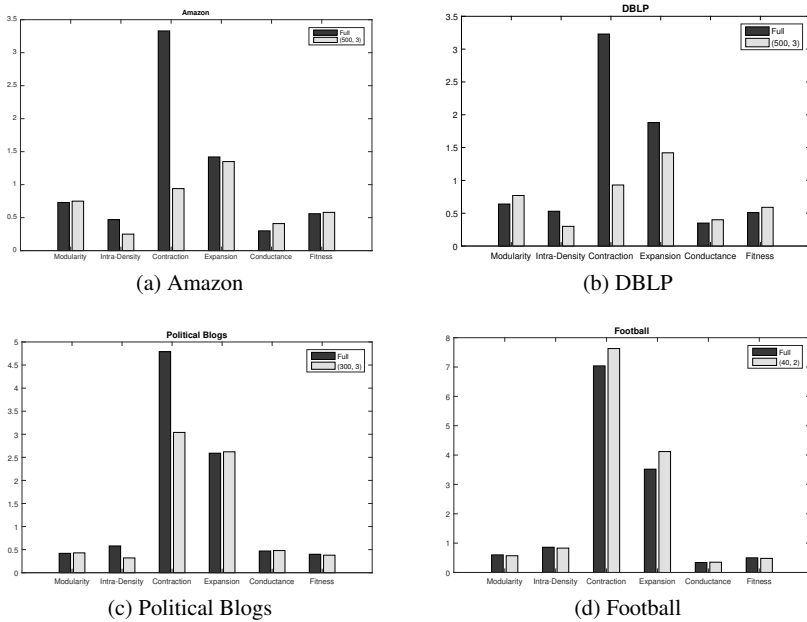


Fig. 5: Performance of Linear Regression

2. **Runtime:** In Fig. 6 it can be seen very clearly that using the algorithms proposed in this paper one can save a tremendous amount of time while preserving the community structure of the graphs. We show the runtime results only for Algorithm 8 to conserve space. Both algorithms perform very similarly. The difference in runtime is very clear for large graphs like Amazon and DBLP. Processing the full Amazon graph requires about 3500ms while the best performing iteration/sampling algorithm takes less than 500ms. This translates to our algorithm being roughly 7 times faster. Similar results can be observed with DBLP and the small datasets.

3. **Centrality:** As it was noted in the summary in Section 1, centrality measures like degree, betweenness and closeness were not as well preserved as the community structure. However, they tend to be closer to the full graph as we increased the number of sampled columns k . For example, $k = 10,000$ on Amazon dataset, for top 10% nodes in terms of degree centrality, yielded an F-measure of 0.02 and 0.002 for $k = 500$. In essence, if one is just interested in getting the community structure of a large graph, with minimal information, then the methodology proposed in this paper produces results of sufficient quality. If more specific features of the graph are required then one would have to invest more time and effort to get more information.

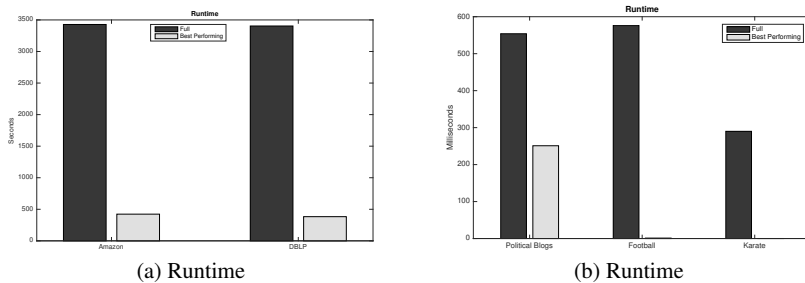


Fig. 6: Runtime of Linear Regression

5 Conclusion

The results presented in this paper show that graphs can indeed be sampled like matrices using sampling techniques from the matrix algebra community while preserving clustering features. We present evidence that using only 0.15% – 30% of the edges of a graph yields communities whose quality is comparable to that of the full graph, according to the most important metrics. We think that going forward, with these results, sampling and reconstruction of large graphs can be considered an important first step before performing machine learning.

Acknowledgements This research was supported by the Army Research Laboratory under Cooperative Agreement W911NF-09-2-0053 (the ARL-NSCTA). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation here on.

References

- [1] Achlioptas, D., McSherry, F.: Fast computation of low-rank matrix approximations. *JACM* (2007)

- [2] Adamic, L.A., Glance, N.: The political blogosphere and the 2004 us election: divided they blog. *Int. Workshop on Link discovery* (2005)
- [3] Arora, S., Hazan, E., Kale, S.: A fast random sampling algorithm for sparsifying matrices. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques* (2006)
- [4] Bader, D.A., Kintali, S., Madduri, K., Mihail, M.: Approximating betweenness centrality. *Algorithms and Models for the Web-Graph* (2007)
- [5] Boutsidis, C., Drineas, P., Magdon-Ismail, M.: Near-optimal column-based matrix reconstruction. *SICOMP* (2014)
- [6] Brandes, U.: A faster algorithm for betweenness centrality. *J. of Math. Sociology* (2001)
- [7] Chen, M., Nguyen, T., Szymanski, B.K.: A new metric for quality of network community structure. *HUMAN* (2013)
- [8] Deshpande, A., Vempala, S.: Adaptive sampling and fast low-rank matrix approximation. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques* (2006)
- [9] Fortunato, S.: Community detection in graphs. *Physics Reports* (2010)
- [10] Gaiteri, C., Chen, M., Szymanski, B., Kuzmin, K., Xie, J., Lee, C., Blanche, T., Neto, E.C., Huang, S.C., Grabowski, T., et al.: Identifying robust communities and multi-community nodes by combining top-down and bottom-up approaches to clustering. *Scientific Reports* (2015)
- [11] Girvan, M., Newman, M.E.: Community structure in social and biological networks. *PNAS* (2002)
- [12] Leskovec, J., Adamic, L.A., Huberman, B.A.: The dynamics of viral marketing. *TWEB* (2007)
- [13] Leskovec, J., Faloutsos, C.: Sampling from large graphs. *ACM SIGKDD* (2006)
- [14] Leskovec, J., Lang, K.J., Mahoney, M.: Empirical comparison of algorithms for network community detection. *WWW* (2010)
- [15] Madduri, K., Ediger, D., Jiang, K., Bader, D., Chavarria-Miranda, D.: A faster parallel algorithm and efficient multithreaded implementations for evaluating betweenness centrality on massive datasets. *IPDPS* (2009)
- [16] Mahoney, M.W.: Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning* (2011)
- [17] Mahoney, M.W., Drineas, P.: CUR matrix decompositions for improved data analysis. *PNAS* (2009)
- [18] Newman, M.E.: Modularity and community structure in networks. *PNAS* (2006)
- [19] Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *PRE* (2004)
- [20] Potamias, M., Bonchi, F., Castillo, C., Gionis, A.: Fast shortest path distance estimation in large networks. *CIKM* (2009)
- [21] Satuluri, V., Parthasarathy, S., Ruan, Y.: Local graph sparsification for scalable clustering. *SIGMOD* (2011)
- [22] Spielman, D.A., Srivastava, N.: Graph sparsification by effective resistances. *SICOMP* (2011)
- [23] Wang, T., Chen, Y., Zhang, Z., Xu, T., Jin, L., Hui, P., Deng, B., Li, X.: Understanding graph sampling algorithms for social network analysis. *ICDCSW* (2011)
- [24] Yang, J., Chen, Y.: Fast computing betweenness centrality with virtual nodes on large sparse networks. *PloS* (2011)
- [25] Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems* (2015)
- [26] Zachary, W.W.: An information flow model for conflict and fission in small groups. *JSTOR* (1977)

Evaluating the community partition quality of a network with a genetic programming approach

Marco Buzzanca, Vincenza Carchiolo, Alessandro Longheu, Michele Malgeri and Giuseppe Mangioni

Abstract Although the problem of partition quality evaluation is well-known in literature, most of the traditional approaches involve the application of a model built upon a theoretical foundation and then applied to real data. Conversely, this work presents a novel approach: it extracts a model from a network which partition in ground-truth communities is known, so that it can be used in other contexts. The extracted model takes the form of a validation function, which is a function that assigns a score to a specific partition of a network: the closer the partition is to the optimal, the better the score. In order to obtain a suitable validation function, we make use of genetic programming, an application of genetic algorithms where the individuals of a population are computer programs. In this paper we present a computationally feasible methodology to set up the genetic programming run, and show our design choices for the terminal set, function set, fitness function and control parameters.

1 Introduction

The community detection problem is not new in the domain of graph theory. The analysis of communities provides a deeper knowledge of the network's structure and the correlation between nodes, which allows the study of the information embedded into networks. Networks concerning healthcare, infection spread, human interactions, economics, transportation, trust and reputation are perfect examples where detecting communities can help to understand the network's structure.

Marco Buzzanca (e-mail: marco.buzzanca@dieei.unict.it)✉ · Vincenza Carchiolo (e-mail: vincenza.carchiolo@dieei.unict.it)✉ · Alessandro Longheu (e-mail: alessandro.longheu@dieei.unict.it)✉ · Michele Malgeri (e-mail: michele.malgeri@dieei.unict.it)✉ · Giuseppe Mangioni (e-mail: giuseppe.mangioni@dieei.unict.it)✉

Dip. Ingegneria Elettrica Elettronica e Informatica (DIEEI), Università degli Studi di Catania, Viale Andrea Doria, 6 - Catania (Italy)

The definition of a community itself is controversial. Intuitively, it can be defined as a set of entities that are close to each other. This notion is quite similar to the concept of *closeness*, which is based on a similarity measure and is usually defined over a set of entities. One of the most acknowledged definitions of community appears in [1]. This definition has given birth to several algorithms for community detection [2] [3] which, for the most part, rely on the optimization of a *validation function* measuring the quality of the community structure. One of the most commonly used functions is the *modularity* function provided by Newman [4]. Approaches based on modularity optimization have however shown some drawbacks, such as the resolution limit introduced in [5], the conjectured hardness described in [6], and the algorithmic infeasibility for large networks or overlapped communities. In this last case, modularity definition has been further extended to tackle overlapping structures [7] [8].

This paper presents a novel approach that attempts to infer the previously mentioned validation function from the network, aiming at obtaining a result that "emerges" from the network, without pre-conditions. Ideally, we want to find a function general enough to properly detect the community structure of several different networks. However, due to the difficulty of the problem, stochastic approaches are often employed to look for near-optimal solutions. One of these stochastic approaches that has been gaining popularity in solving these kind of problems is the *Genetic Programming* (GP) method. GP is a branch of evolutionary computation, and can be seen as an application of the more well-known genetic algorithms. The main difference is that the individuals of a population are not strings of bits but computer programs made of constants, variables, and functions. These pieces of code are different for each individual of a population, much like each organism has different genes compared to other individuals of the same species.

Although the idea of making computer automatically solve problems is not new [9] [10], only recently the technological advancements in the field of computing speed allows to exploit these techniques to solve more complex cases, including community detection. Evaluating the community partition quality via GP consists in finding an individual which can be used as a validation function that allows us to evaluate partitions of a network. Of course we would like this function to be as general as possible, in order to apply it to different networks and still produce reasonable results. But it is also possible that the application of GP to different networks could lead to different functions, therefore the question is whether a function that minimizes the difference among the set of functions related to different networks would exist and how it could be found. In this paper we show how to build a validation function that is computationally feasible, and how to apply GP in order to solve the problem. In particular, in sec. 2 an overview of GP is presented, while its application is illustrated in sections 3 and 4. Final considerations, together with further works are presented in sec. 5.

2 Genetic Programming

The idea of having machines automatically solve problems has always been central in the domain of artificial intelligence. A relevant problem since the early days of artificial intelligence is however a machine would solve a problem which solution is a computer algorithm itself.

GP attempts to take on such challenge by making use of the concepts of evolutionary computation, which borrows from nature the idea of the survival of the fittest. It aims at generating a feasible algorithm that can solve the specified problem without requiring the user to specify the shape of the solution in advance.

The gist of GP consists in evolving a population of computer programs. Computer programs which participate in the process are named *individuals*. At each iteration of the process, the population is evaluated, and each individual is given a numerical score named *fitness*. The better the fitness, the more likely an individual is a solution to the GP problem. The fitter individuals are then manipulated by the use of *genetic operations* in order to generate a better population for the next iteration. The process continues until an exit condition is satisfied: the fittest individual that was ever bred among all the iterations will be designated as the solution to the problem. This whole process is shown in Figure 1.

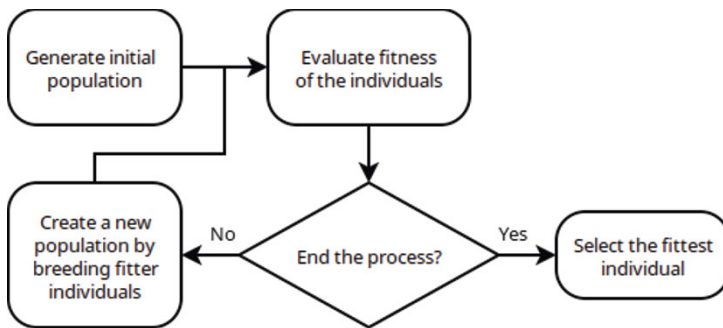


Fig. 1: Overview of the Genetic Programming process

The GP process is inherently random, and sometimes it produces no meaningful solutions. However, this randomness allows GP to avoid the traditional pitfalls of deterministic search algorithms.

Setting up a GP problem means specifying how an individual is constructed in terms of terminals and functions, defining a proper fitness function and providing parameters that control the run, including the exit conditions, as summarized in Figure 2.

2.1 Terminal and function sets

The terminal set is the set of values that are used as arguments of the functions in the function set. It may consist of:

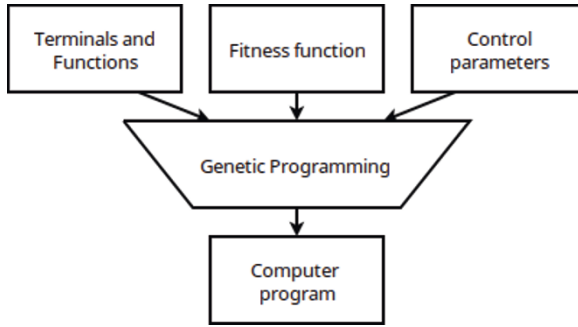


Fig. 2: Preparatory steps for the Genetic Programming process

- External inputs of the program, typically represented by named variables like x .
- 0-ary functions, like `time()`, that may return a different value each time they are run.
- Constants, either determined before the run or created by mutation.

The function set is very dependent on the application domain. In simple numerical problems, it may consist of the four basic arithmetic functions (+, -, *, /), but they could be higher level functions: for example, if we are looking for an auto-pilot system for a car, functions could include `steer()`, `accelerate()`, `decelerate()` in a simplest case.

2.2 *Fitness function*

Defining a good fitness function is perhaps the most crucial step when setting up a GP problem. A good fitness function should always return large (small) values for individuals that fit, and small (large) values for individuals less fit, so that the individual which has the highest (lowest) score is the fittest. It is often the sole mechanism to provide a high-level statement of the problem's requirements. For example, if the GP problem consists in finding the closest rational number for any real number x , the program `floor(x*100)/100`, is more fit than the program `floor(x*10)/10`, as it gives a more accurate result for all values of x , so it should receive a better score.

2.3 *Control parameters*

At last, there are several parameters that need to be configured in order to start the GP search: the termination criterion, the population size, how the initial population is created, the probability of applying a genetic operators and so on. Of all these parameters, the most important two are the population size and the termination criterion. Regrettably, it is not possible to make general recommendations regarding

an optimal set of GP parameters, as it strictly depends on the specific application. However, GP is often robust, and many different parameter values may work.

3 Community structure validation problem

As specified in the introduction, the goal of this paper is to attempt to solve the problem of community structure validation with a GP approach. The solution takes the form of a *validation function*, which is a function that assigns a certain score to a partition of a network in clusters: the closer to the optimal that partition, the better the score. For simplicity's sake, we will say that a partition is better (worse) than another when it's closer (farther) from the optimal partition.

Note that in the general case we cannot assume that a better partition always gets an higher score compared to a worse partition. The validation function might assign a higher score to worse partitions, depending on its shape. This is the reason why we will not use the terms "lower" or "higher" when considering the validation function score, rather the more generic "better" and "worse".

Such a function could be used in conjunction with global optimization methods to find communities: in this case, we want to find the partition that yields the best score, or get reasonably close to that. However, this is beyond the scope of this paper; for the time being, it is necessary to first determine if a solution to the problem exists. Let's first describe in more formally what we are looking for.

Let's assume we have an undirected, unweighted network $G = (V, E)$, where V is the set of nodes and $E \subseteq V \times V$ is the set of edges. If we name \mathcal{P} the space of all the possible partitions of G , we are looking for a validation function $\beta : \mathcal{T}_{\mathcal{P}} \rightarrow \mathbb{R}$ that has a reasonable computational complexity. $\mathcal{T}_{\mathcal{P}}$ is the space of terminal sets obtained from all the partitions in \mathcal{P} : its generic element is simply the terminal set of a specific partition $P \in \mathcal{P}$. However, the β function is very difficult to handle as an individual of the GP problem due to the high dimensionality of $\mathcal{T}_{\mathcal{P}}$, as it would require many terminals to provide enough information to describe the whole partition of the network.

Instead of looking for a validation function as a whole, to reduce the dimensionality of the selection function, we decided to fragment the function β so that it operates on the terminal sets $T_e \in \mathcal{T}_E, \forall e \in E$, which have less dimensions:

$$\beta(T_e) = \sum_{i=1}^{r_k} \sum_{e \in E_i} f(T_e) \quad (1)$$

$f : \mathcal{T}_E \rightarrow \mathbb{R}$ is an individual of the population in the GP problem, and \mathcal{T}_E is the space of terminal sets obtained from all the edges in E , hence a generic $T_e \in \mathcal{T}_E$ is the terminal set of an edge $e = (v, w)$. This terminal set should contain numeric information about the nodes that connects. This includes microscopic parameters like the degree of v and w or their structural difference. However, there are also mesoscopic (related to the communities e belongs to) and macroscopic parameters (related to the whole network) that are worth considering even when evaluating the score of a single edge. For example, it may be worth comparing the degree of v or

w to the average degree of the nodes within the same community, or to the average degree of the nodes within the network. Of course we don't know exactly how the terminals will be compared within the function f due to the nature of GP, but we believe that the terminal set must offer the opportunity for such comparisons to happen.

Finally, note that we are excluding on purpose all $(v, w) \in E : v \in C_i, w \in C_j, C_i \neq C_j$. This simplification is necessary to further reduce the dimensionality of the terminal sets and the overall complexity of the GP problem, because including these edges would imply two problems to be addressed:

- It would be necessary for the f to behave differently for inter-community and intra-community links. This makes the search much harder, so, as far as complexity is concerned, it is better that all the edges are of the same type.
- Inter-community links require more terminals than intra-community links, because they bear mesoscopic information about two communities instead of one.

In conclusion, our GP problem consists in finding an individual $f : \mathcal{T}_E \rightarrow \mathbb{R}$ that, applied to all intra-community edges, will provide a score to a certain partition $P \in \mathcal{P}$. The terminal set will provide access to microscopic, mesoscopic and macroscopic properties that can be used by the GP algorithm to create a suitable f . In the following section we describe in detail parameters of the GP process.

4 Methodology

In this section the parameters that characterize the proposed GP run are illustrated. As already mentioned in section 2, these are the terminal set, the function set, the fitness function, and all the control parameters such as the population size and the termination criterion.

4.1 Terminal set

The terminal set was one of the most challenging parameters to define. On one hand, we want to include several different properties from the network at different levels (microscopic, mesoscopic, macroscopic), on the other hand too many properties would raise the complexity of the GP problem, making the solution harder to search for. We already mentioned in section 3 that we simplified the original problem in order to make use of a reduced terminal set. If we name the generic edge $e_i = (v_i, w_i) \in E_i$ belonging to the community C_i , the terminal set we decided to make use of is the following:

- Microscopic Parameters
 - degree of node v_i ;
 - degree of node w_i ;
 - structural equivalence between v_i and w_i ;

- number of edges of v_i that point to other nodes in C_i ;
- number of edges of w_i that point to other nodes in C_i .
- Mesoscopic Parameters
 - average degree of nodes in C_i ;
 - total number of edges in C_i ;
 - total number of nodes in C_i .
- Macroscopic Parameters
 - average degree of nodes;
 - total number of edges;
 - total number of nodes.

The information concerning each level is very abstract and simple by design: we don't want to bias the GP run with excessively refined mathematical models. The only exception to this could be the structural equivalence, which is computed via the cosine similarity. If the results suggest that the quality of the solution would benefit from a larger terminal set, it is of course always possible to add other parameters in subsequent runs. We could also remove some of the parameters if we see that they come out unused in the fitter individuals.

4.2 Function set

Contrary to the terminal set, the function set is small, and consists of only five functions: $\{+, -, \times, \div, \sqrt{\cdot}\}$, which are the binary addition, subtraction, multiplication and *protected division* and the unary square root operator. The protected division operator \div is defined as:

$$a \div b = \begin{cases} 1, & \text{for } b = 0 \\ \frac{a}{b}, & \text{otherwise} \end{cases}$$

The function set is small for two reasons: having a smaller function set decreases the complexity of the algorithm, and since most of the used functions are simple, they have less impact on the overall computation time. Note that there are important terminals and functions that can be derived from a combination of elements from the defined terminal and function sets:

- 0 can be written as $n - n, \forall n \in \mathbb{R}$;
- 1 can be written as $n \div (n - n), \forall n, m \in \mathbb{R}$;
- n^2 can be written as $n \times n, \forall n \in \mathbb{R}$;
- $|n|$ can be written as $\sqrt{n \times n}, \forall n \in \mathbb{R}$.

4.3 Fitness

Determining a proper fitness function is also a major challenge, and often the success of a GP search depends on how accurately the fitness functions validates the correct solution. In our case, the fitness function needs to evaluate how well our validation function β (1) behaves. In practice, its behavior is tested by applying it to the ground-truth partition P^* and d randomly generated partitions. The scores of these randomly generated partitions are then compared against the score of the ground-truth partition. Intuitively, the more a partition P_k is similar to the ground-truth partition, the better score β should yield.

Unfortunately, comparing the scores as they are gives little or no information about how accurate is the validation function in scoring a specific partition. Assuming P^* is the ground-truth partition, how can we say that $\beta(T_{P^*})$ yields the best score if we don't know the maximum value that β can assume? Assuming P is a generic partition, how can we say that $\beta(T_{P_k})$ is better or worse than $\beta(T_{P^*})$ when we don't know the shape of β ? This is why we decided to measure the correlation between the difference of the two β scores and the normalized mutual information (NMI) a measure of how different two partitions are. If the difference is correlated to the NMI, it means that the β function behaves as desired, and it is a good candidate for our solution. Note that we could use any kind of difference measurement: we chose NMI because it is well-studied and has convenient properties [11].

We show how to apply the aforementioned intuitions in order to measure this correlation and obtain our fitness function φ . First, let's define the basic building block for our fitness function, which is the function $\gamma: \mathcal{P} \rightarrow \mathbb{R}$, defined as following:

$$\gamma(P) = \frac{|\beta(T_{P^*}) - \beta(T_P)|}{NMI(P^*, P)} \quad (2)$$

This function alone does not measure correlation of course. To do that, we need to consider its *standard deviation* σ_γ :

$$\mu_\gamma = \sum_{i=1}^d \frac{\gamma(P_i)}{d} \quad \varphi(f) = \sigma_\gamma = \sqrt{\sum_{i=1}^d [\gamma(P_i) - \mu_\gamma]^2} \quad (3)$$

Given the definition of our fitness function, we may conclude that the best individual f is the one that minimizes φ .

4.4 Control parameters

Compared to the other settings, determining the optimal control parameters beforehand is usually not possible. Things like population size, crossover ratio, number of generations, are best determined via experimentation. As far as the complexity parameters are concerned, in principle, we start with a small population (about 50 individuals) and an average number of generations (about 30). These numbers may be refined according to the performance of the GP framework in terms of the quality of the results and computing time.

The crossover ratio, which is the chance that crossover occurs between two genes, is also an important factor. Normally, each generation is subject to different genetic operators randomly. Certain individuals will undergo crossovers, others mutation. The crossover ratio indicates what is the chance of two individuals to crossover. A traditional approach [12] is to have a crossover ratio of 0.9, while the mutation ratio is set to the remaining 0.1, and we believe this is a good starting point for our experiment.

There is also a variety of different genetic operators and strategies that have to be chosen. For example, it makes sense for certain GP problems to adopt automatically defined functions (ADF), a way to evolve reusable components, but they are most effective in problems which present some degree of regularity. Also, there are many different kinds of crossover and mutation operators [13], and the problem of determining which kind of operator to use is complex [14].

At first, the selection of operators will probably be limited to what the GP framework has to offer. Then, if the results suggest that the GP problem could benefit from the application of specific operators that are not implemented in the framework, we may eventually extend the framework by adding the missing operators, or migrate to a different one.

In conclusion, it is hard to fully specify what control parameters to use without experimentation. We will proceed using general recommendations about their values, then we will progressively refine the selection with the feedback obtained from previous runs.

5 Conclusion

This work presents a novel methodology for community structure validation that makes use of Genetic Programming (GP). First we described the problem in general, specifying how can it be treated as a GP problem. The idea is to find a validation function β that can assign a score to partitions of the network. However, it is not possible to set up the GP run using a population of validation functions, as it would make the computation time too long. Hence, we decided to fragment β so that it can operate on terminal sets with smaller dimensions, reducing the overall complexity.

We also presented a list of viable parameters for the GP run. We used microscopic, mesoscopic and macroscopic properties of the network to build the terminal set. We choose few operators for the function set, in order to keep the complexity at minimum. We designed the fitness function so that it measures how a generic validation function resulting from a step of the GP run compares to the well-known normalized mutual information.

In the future we aim at putting in practice the described proposal, and to benchmark our validation function against networks which have a known partition in communities.

References

- [1] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, p. 026113, 2004.
- [2] L. Danon, J. Duch, A. Diaz-Guilera, and A. Arenas, "Comparing community structure identification," 2005.
- [3] S. Fortunato and C. Castellano, *Community Structure in Graphs*. Encyclopedia of Complexity and System Science, Springer, 2008.
- [4] M. E. J. Newman, "Modularity and community structure in networks," *PROC.NATL.ACAD.SCI.USA*, vol. 103, p. 8577, 2006.
- [5] S. Fortunato and M. Barthelemy, "Resolution limit in community detection," *PROC.NATL.ACAD.SCI.USA*, vol. 104, p. 36, 2007.
- [6] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner, "On modularity clustering," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 20, pp. 172–188, Feb. 2008.
- [7] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communities," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, no. 03, p. P03024, 2009.
- [8] V. Carchiolo, A. Longheu, M. Malgeri, and G. Mangioni, "Search for overlapped communities by parallel genetic algorithms," *International Journal of Computer Science and Information Security*, Vol. 6 No. 2, pp. 113-118, vol. abs/0912.0913, 2009.
- [9] R. Poli, W. B. Langdon, and N. F. McPhee, *A field guide to genetic programming*. 2008.
- [10] W. Banzhaf, F. D. Francone, R. E. Keller, and P. Nordin, *Genetic Programming: An Introduction: on the Automatic Evolution of Computer Programs and Its Applications*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998.
- [11] J. Wu, H. Xiong, and J. Chen, "Adapting the right measures for k-means clustering," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, (New York, NY, USA), pp. 877–886, ACM, 2009.
- [12] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection*. Cambridge, MA: MIT Press, 1992.
- [13] R. Poli, W. B. Langdon, and N. F. McPhee, *A field guide to genetic programming*. March 2008.
- [14] S. Luke, "A comparison of crossover and mutation in genetic programming," in *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pp. 240–248, Morgan Kaufmann, 1997.

A graph-based meta-approach for tag recommendation

Manel Hmimida and Rushed Kanawati

Abstract In this paper we propose a graph-coarsening approach that aims to speed-up the execution time of graph-based tag recommenders in large-scale folksonomies. A community detection algorithm in *multiplex* networks is applied for coarsening the hypergraph depicting a folksonomy. Experiments on real datasets show the validity of the approach.

1 Introduction

Social tagging systems, or folksonomies, are popular Web 2.0 tools that allow people to share and organize large sets of resources such as bookmarks, documents, photos, etc. Tag recommendation is a core service in such systems [1, 2]. The goal is to compute the most adequate tag set that a user can apply to annotate a given resource. This helps in controlling the tag vocabulary set, enhancing hence its usefulness for resource access and searching while keeping the annotation process user-centred. This problem has attracted much of interests in the last few years with a variety of different approaches being proposed [3, 4, 5, 6, 7, 8]. Graph-based approaches constitute a major trend in this area [9, 10, 11]. These are attractive approaches since they rely only on mining the induced graph structure of the tagging history making these independent from the type of annotated resources [5, 6, 7, 8, 12, 13, 14]. Actually, the tagging activity history can be represented as a 3-uniform hypergraph where all hyperedges involve three nodes of different types: a user, a resource and a tag.

While graph-based approaches yield interesting results, they often suffer from high execution times due to the large-scale of handled graphs. In this work, we propose a graph-coarsening based approach that can overcome this drawback. The proposed approach is decomposed into two steps: an *offline* step where the folkson-

Manel Hmimida (e-mail: rushed.kanawati@lipn.fr) · Rushed Kanawati (e-mail: manel.hmimida@lipn.fr)
LIPN UMR CNRS 7030, SPC-UP13

omy hypergraph is abstracted by applying a topological clustering approach to the three sets of nodes: users, resources and tags, and an online step during which recommended tags are computed. Upon receiving a query composed of a target user and resource we apply a basic graph-based tag recommendation approach to the abstract graph in order to compute a set of recommended abstract tags. These will be used to construct a new reduced graph, called the *contextual graph* by unfolding the abstract subgraph composed of the set of recommended abstract tags and nodes representing the cluster of users (resp. resources) to which the target user (resp. resource) belongs to. Again the same basic graph-based tag recommendation approach is applied to this new reduced graph in order to compute the final set of tags to recommend. Thus the approach consists in replacing the execution of a standard graph-based tag recommendation approach on a large-scale graph by two executions of the same approach on two reduced graphs. This is expected to drastically reduce the online recommendation computation time. The quality of computed recommendations is also expected to be enhanced since the contextual graph is focused on the query (target user and resource) avoiding taking into account query-irrelevant data. Main contributions of this work are :

- Defining a meta topological approach that can be applied to enhance graph-based tag recommendation approaches in terms of execution time and recommendation quality.
- Benchmarking different community detection algorithms for multiplex networks in the context of tag recommendation problem.
- Providing experiments on real dataset.

The remainder of this paper is structured as follows. In section 2 we provide a quick survey on main topological approaches for tag recommendation. The proposed approach is detailed in section 3. Experiments on real networks are reported and commented in section 4. Finally, conclusions are drawn in section 5.

2 Related work

A folksonomy can be formally, represented as a 3-uniform hypergraph $G = \langle V = U \cup R \cup T, Y \subseteq U \times R \times T \rangle$ where V is a set of nodes composed of three disjoint sets : U set of users, R set of resources and T set of tags. Y is a ternary relationship such that : $(u, r, t) \in Y$ if user $u \in U$ has annotated resource $r \in R$ using the tag $t \in T$. A graph-based, or a topological, tag recommender relies only on mining the structure of the hypergraph in order to infer the top- k tags that are the most relevant to be used by a given user to annotate a given resource. Existing topological approaches can be classified into four main classes:

- Node ranking based approaches [6, 12, 15, 16, 17]
- Link prediction based approaches [8, 10].
- Graph search based approaches [11]
- Clustering-based approaches [5, 7]

In this work we go steps further by first proposing a graph coarsening approach based on clustering all the three sets of involved nodes (i.e. users, resources and tags) each based on similarities in function of their relationships to both other types of nodes. Then, a set of recommended cluster of tags is computed on the coarsened graph. This intermediate result is used to extract a contextual graph from the raw graph of the folksonomy that is focused on the target user and resource (i.e. the query). A graph-based recommendation approach applied to this later graph in order to compute the final list of tags to recommend. The proposed approach is detailed in next section.

3 Proposed approach

Algorithm 10 sketches the outlines of the proposed tag recommendation approach. In order to treat a recommendation query $q = (u, r)$ defined by a couple of target user u and resource r , the approach requires the following inputs: G the raw graph of the folksonomy, G_c : a compression of G and graph-based tag recommendation approach: **tagRecommender()**. The later can be any of the graph-based tag recommendation approaches mentioned in section 2

Algorithm 10 Tag recommendation approach

Require: $q = (u, r)$ #user query

$G = \langle (U, R, T), E \rangle$ # a folksonomy graph

$G_c = \langle (C_U, C_R, C_T), E_c \rangle ::= \mathbf{Compression}(G)$

tagRecommender() # a graph-based tag recommender

Ensure: K_t : recommended tags

1: $C_u^t \leftarrow c \in C_U : q.u \in c$

2: $C_r^t \leftarrow c \in C_R : q.r \in c$

3: $\hat{q} \leftarrow (C_u^t, C_r^t)$

4: $K_c \leftarrow \mathbf{tagRecommender}(\hat{q}, G_c)$

5: $G_{context} \leftarrow \mathbf{induced_subgraph}(G, C_u^t, C_r^t, K_c)$

6: $K_t \leftarrow \mathbf{tagRecommender}(q, G_{context})$

Return: K_t

The folksonomy graph compression is done off-line. Upon receiving a recommendation query $q = (u^t, r^t)$, the algorithm starts by rewriting the query q in terms of clusters of users (denoted C_u^t) and cluster of resources (denoted C_r^t) computed during the graph compression process (line 1 to 3). This step shows clearly one classical limit of pure graph-based approaches which is the *cold start* problem: no recommendation can be computed for new users and/or new resources. Let \hat{q} be the rewritten query. The tag recommender function is then applied to the compressed graph G_c in order to handle \hat{q} . A set K_c of recommended cluster of tags is then obtained. These will be used along with C_u^t, C_r^t to extract from the raw graph G a *contextual graph*

$G_{context}$ defined as subgraph of G defined over the set of nodes in $C_u^t \cup C_r^t \cup K_c$. The same graph-based recommendation function is applied to the contextual graph in order to get the final set of tags to recommend. The tag recommendation process consists then on applying the same recommendation approach twice on two reduced graphs. This allows reducing the execution time of the whole approach (without taking into account the time for compressing the raw graph which is done off-line). The computation of the contextual graph is also expected to enhance the performances of the recommendation approach since it is expected to avoid taking into account irrelevant nodes.

A central step in the approach is the graph compression one. The principal of graph compression is to compute clusters over the three sets of nodes : users, resources and tags. The graph coarsening consists then in replacing each cluster of nodes by one *abstract* node. Clustering algorithms requires defining a *dissimilarity function* over the set of items to cluster. But since we are targeting pure topological (or graph-based) recommendation approach the only information we can harvest over nodes is their connectivity to other nodes. We use this information to infer relations between nodes of each type in function of their connectivity to nodes of the other two types. This is simply done by projecting the raw folksonomy graph on each of its mains components : users, resources and tags. For each type of nodes we can then infer two types of relations in function of the other two node's types. This allows then to define three *multiplex networks*, each composed of two layers. A multiplex network is defined as a multi-layer network where each layer is composed of the same set of nodes. Each layer defines a set of different links between nodes [18]. One way to define clusters over each set of nodes consists then in applying a community detection algorithm to each of the obtained multiplex networks. A wide variety of community detection algorithms for multiplex networks has been recently proposed in the scientific literature [19]. A brief review of main approaches is presented in next section.

4 Community detection in multiplex networks

A multiplex network G is defined as triplet $G = \langle V, E, C \rangle$ where:

- V is a set of nodes,
- $E = \{E_1, \dots, E_\alpha\} : \forall k \in [1, \alpha] E_k \subseteq V \times V$ a set of different α layers; each defining a different type of relation between nodes in V
- C a set of inter layer *coupling* links. Different coupling schemes can be defined. Basic schemes include *ordinal coupling* and *categorical coupling*. The first consists on linking nodes between adjacent layers while the later consists in linking each node to itself from each layer to every other layer.

In our case, where we have only two-layer multiplex networks, ordinal and categorical coupling are roughly the same. The problem of community detection in complex networks is about finding dense subgraphs that are loosely coupled. The concept of community in a complex network is still fuzzy in spite of the huge number

of papers that have been published in this field in the last few years [20]. Defining a community in a multiplex network is even worse since we need to define what is a dense subgraph in a multiplex network [21]. Despite this difficulty, an increasing number of work has been proposed in the last few years to deal with this problem. Existing approaches can be broadly classified into two distinct classes: the first class regroups work that consist in transforming the problem of community detection in multiplex networks into the one of commuting communities in a *monoplex* network. The second class of work regroups algorithms that generalize existing algorithms to the case of multiplex networks.

Trivial approaches from the first class are, *layer aggregation* (LA) and *partition aggregation* (PA) approaches. The first consists in simply aggregating layers of a multiplex network in one layer. A classical community detection algorithm can then be applied to the aggregated network. Different aggregation schemes can be applied. In general, the layer aggregation approach consists on transforming a multiplex network into a weighted monoplex graph $G = \langle V, E, W \rangle$ where W is a weight matrix. Different weighting scheme have been proposed including linear combination [22] and similarity-based aggregation [23].

The principle of partition aggregation approaches is to apply a community detection algorithm to each layer aside then to combine resulting community structures into one clustering. This can be made by applying any *ensemble clustering* approach [24, 25, 26, 27].

More interestingly, algorithms that extend existing one to the multiplex network settings have been proposed in different work in the last few years. In [28] a generalization of the modularity function has been proposed. This is given by:

$$Q_{multiplex}(P) = \frac{1}{2\mu} \sum_{c \in P} \sum_{\substack{i, j \in c \\ k, l: 1 \rightarrow \alpha}} \left(\left(A_{ij}^{[s]} - \lambda_k \frac{d_i^{[k]} d_j^{[k]}}{2m^{[k]}} \right) \delta_{kl} + \delta_{ij} C_{ij}^{kl} \right)$$

where $\mu = \sum_{\substack{j \in V \\ k, l: 1 \rightarrow \alpha}} m^{[k]} + C_{jkl}$ is a normalization factor, and λ_k is a resolution factor

as introduced [29] in order to cope with the modularity resolution problem. Note that in our case, inter-layer links are implicit links connecting node i to itself in the others layers. Therefore we have: $C_{ij}^{kl} = 0 \forall i \neq j$.

By using this *multiplex modularity*, algorithms that apply greedy modularity optimization can be directly applied to multiplex networks. An example is the *Gen-Louvain* algorithm [28] that is the generalization of the well-known *Louvain* algorithm [30]. Both *WalkTrap* [31] and *InfoMap* [32] algorithms have been generalized to multiplex setting in respectively [33] and [34].

The algorithm *Mux-Licod* proposed in [19] is a generalization of the seed-centric algorithm *Licod* proposed in [35]. We give hereafter some details about this algorithm since it is used later in experiments reported in this paper. A survey on seed-centric community detection algorithms is provided in [36]. The following algorithm gives the general outlines of a seed-centric approach.

Algorithm 11 General seed-centric community detection algorithm

Require: $G = \langle V, E \rangle$ a connected graph,

- 1: $\mathcal{C} \leftarrow \emptyset$
 - 2: $S \leftarrow \text{compute_seeds}(G)$
 - 3: **for** $s \in S$ **do**
 - 4: $C_s \leftarrow \text{compute_local_com}(s, G)$
 - 5: $\mathcal{C} \leftarrow \mathcal{C} + C_s$
 - 6: **end for**
- Return:** $\text{compute_community}(\mathcal{C})$

The idea is to compute seeds in the network: nodes or subgraphs that play central role in the network. Then local communities centred on these seeds are computed. Lastly, the set computed local communities are used to infer a community structure over the whole network. The *Licod* algorithm applies roughly the same scheme. Seeds are selected to be nodes that have higher centrality degree than most of direct neighbors. Once seeds are detected, each node in the network rank the list of seeds according to its local preference to be a member in the seed's local community. This is simply done by ranking the list of seeds in function of the length of shortest path linking the node to it the seed. Ties are broken randomly. Then, each node exchange with direct neighbors its preference list. Nodes merge preferences of neighboring nodes, using a classical preference merging algorithm [37]. These two steps of exchanging preference and merging preference is iterated till stabilization. Each node will be then assigned to the community of the top ranked seed in the local ranked list of seeds. Extending this algorithm to cope with multiplex networks is straightforward. The following concepts should be defined for a multiplex network before applying the algorithm: The degree centrality, the length of the shortest path between two nodes and the neighborhood set of a node. The following definitions are applied in the context of the *Mux-Licod* algorithm [19]: The multiplex degree centrality of a node i is computed by the following formula proposed initially in [38]:

$$d_i^{\text{multiplex}} = - \sum_{k=1}^{\alpha} \frac{d_i^{[k]}}{d_i^{\text{[tot]}}} \log \left(\frac{d_i^{[k]}}{d_i^{\text{[tot]}}} \right) \quad (1)$$

where $d_i^{[k]}$ is the degree of node i in layer k and $d_i^{\text{[tot]}} = \sum_k d_i^{[k]}$.

The multiplex shortest path is defined by :

$$SP(i, j)^{\text{multiplex}} = \frac{\sum_{k=1}^{\alpha} SP(i, j)^{[k]}}{\alpha} \quad (2)$$

where $SP(i, j)^{[k]}$ is the length of the shortest path between nodes i, j in layer k . Finally, the multiplex neighborhood of a node i is defined by:

$$\Gamma^{\text{mux}}(i) = \{x \in \Gamma(i)^{\text{tot}} : \frac{\Gamma(i)^{\text{tot}} \cap \Gamma(x)^{\text{tot}}}{\Gamma(i)^{\text{tot}} \cup \Gamma(x)^{\text{tot}}} \geq \delta\} \quad (3)$$

where $\Gamma(i)^{\text{tot}} = \cup_k \Gamma(i)^{[k]}$. $\Gamma(i)^{[k]}$ is the neighborhood of i in layer k . $\delta \in [0, 1]$ is a similarity threshold. This formula states simple that the multiplex neighborhood of

a node i is composed of a subset of neighbors of i in all layers that have a Jaccard similarity above δ .

5 Experiments

We evaluate the proposed tag recommender on two benchmark datasets extracted from two folksonomies: Bibsonomy a bibliographical reference sharing system and *Deliciousa* social bookmark sharing system. These datasets have been provided in the context of the *HetRec 2011* competition [39]. Each dataset is composed of a set of triadic relationships : connecting a user, a resource and a tag. Table 1 give basic information about both used datasets.

Dataset	U	T	R	# Edges
Bibsonomy	116	412	361	24 297
Delicious	1 867	53 388	69 226	437 593

Table 1: Basic statics describing used datasets

Each dataset is divided into a learning set and a testing set. The size of the test set is taken to be 5% of the whole dataset. The performances of the recommender system is evaluated in function of precision of provided recommendations and the overall execution time. The graph compression time is not included in the reported execution time.

Let $q = (u, r)$ be a recommendation query in the test set. Let $T(u, r)$ be the set of tags associated to the couple (u, r) in the test set. Let $\tilde{T}(u, r)$ be the set of tags returned by the tag recommender system in response to the query $q = (u, r)$. The precision for the query q is then given by the following formula:

$$Pr(q = (u, r)) = \frac{|T(u, r) \cap \tilde{T}(u, r)|}{|\tilde{T}(u, r)|} \quad (4)$$

Thus the computed precision does not take into account the order in which tags are recommended.

The proposed tag recommender has the following parameters :

- The community detection algorithm to apply for graph compression. Two community detection algorithms are selected, the well known *Louvain* approach [30] and *Licod* [35]. Both algorithms are used in combination with layer aggregation (denoted $LA()$) and ensemble clustering (denoted $EC()$) and in their respective generalized versions to multiplex networks : *GenLouvain*[28] and *Mux-Licod* [19]. All these approaches are implemented *MUNA* a Multiplex network analysis package developed in *R* [40].
- *tagRecommender()*: the basic topological tag recommendation approach to use. *FolkRank* is applied as a basic tag recommender [6].

- The number of tags to recommend after applying *tagRecommender* to the extracted contextual graph. This is denoted $|K_r|$. We make vary $|K_r| \in [1, 4]$ since most resources have up to 4 tags in both datasets.
- The number of abstract tags to retain after applying *tagRecommender* to the compressed graph. This is denoted by $|K_c|$. For each value of $|K_r|$ we vary $|K_c| \in [1, 5]$.

Figure 1a (resp. 1b, 1c, 4) shows the variation of the obtained precision in function of the number of retained clusters of tags (i.e. $|K_c|$) when we limit the number of tags to recommend (i.e. $|K_r|$) to 1 (resp. 2, 3, 4).

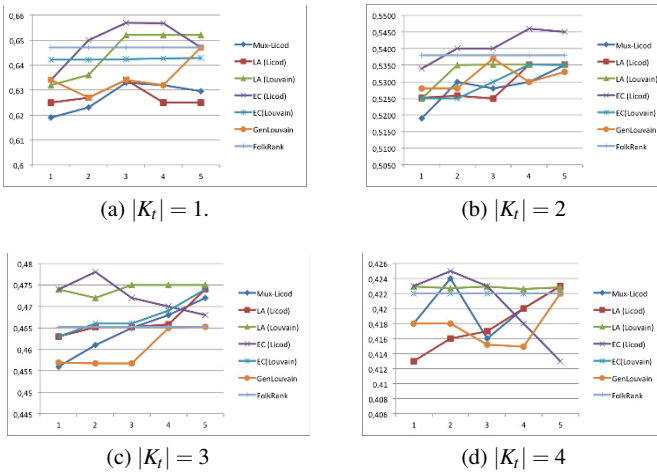


Fig. 1: Variation of average precision in function of $|K_r| = 4$ and $|K_c|$ on Bibsonomy dataset

Figures 2a, 2b, 2c, 2d show the results of the same above mentioned configuration of the tested approach when applied to the Delicious dataset.

First, we notice (across all figures) that the average precision decreases when $|K_r|$ increases. To illustrate this inverse relationship between precision and $|K_r|$ we plot on figure 3 the obtained average precision when applying the *Mux-licod* community detection algorithm on both datasets, while varying $|K_r|$ from 1 to 4 and fixing $|K_c|$ to 2. Similar trends can readily be figured out for other configurations (through figures 1a to 2d).

This observation is also true for the basic *FolkRank* approach which yields precision of 0.65 for $|K_r| = 1$ and drops to 0.423 when $|K_r|$ is set to 4. This may mean that the node ranking approach ranks mostly true positive recommendations more frequently at the top of the result list.

Next, in almost all configurations, the precision (slightly) increases when $|K_c|$ increase. This is particularly true when using *Licod* or *Mux-Licod* for graph compression. The use of *Louvain* is less sensitive to this issue. This is due to the fact

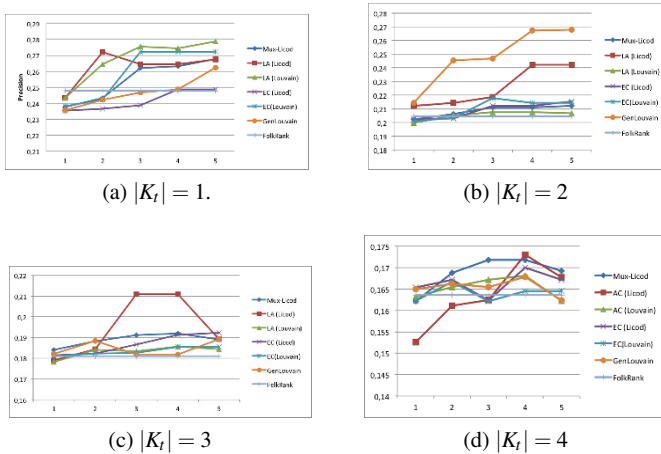


Fig. 2: Variation of average precision in function of $|K_c|$ and $|K_t|$ on Delicious dataset

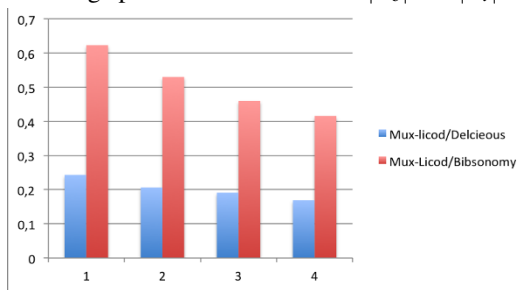


Fig. 3: Variation of average precision in function of $|K_t|$ using Mux-Licod and fixing $|K_c|$ to 2

that Louvain returns usually huge communities (since it is based on optimizing the modularity objective function). We notice also that for almost all configuration the precision increases when increasing $|K_c|$ from 1 to 2. The number of retained clusters of tags influence the size and the quality of the extracted contextual graph. It is clear that there is a trade-off to find between having a very focused contextual graph that might not include relevant tags to recommend and a more larger graph that will increase also the noise level in the set of tags to explore. $|K_c| = 2$ seems to be a good option. We notice also that in all configurations, the graph coarsening approach yields slightly better results than the basic *FolkRank* approach. The gain in precision seems not to be significant (using FolkRank as a basic tag recommender algorithm), but at least there is no drop in the quality of obtained recommendations despite the compression process (that leads naturally for some information loss). More important are the results in terms of execution times. Next two figures show the on-line execution times for treating test queries for each dataset.

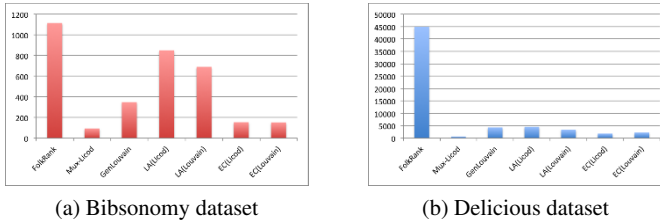


Fig. 4: Recommendation computation time

Both figures 4a, 4b show that our approach allows a drastic reduction of the execution time, compared to the *FolkRank* approach applied directly to the raw graph. For both datasets, the best execution time is obtained using the *Mux-Licod*. This is clearly an encouraging result. Investigations should be continued, mainly by exploring using other basic graph-based tag recommender in order to enhance both recommendation quality and keeping execution time as low as we've already obtained.

6 Conclusion

In this paper, we have proposed a graph-coarsening based meta approach for tag recommendation computation. A core component of the approach is a community detection algorithm in multiplex networks. Experiments on real-world data shows that the proposed approach allows decreasing drastically the recommendation computation time without affecting the quality of obtained recommendations. This approach can be also used as a benchmark for comparing different community detection algorithms in multiplex network. It provides a clear application-driven evaluation of community detection algorithms. Seed-centric approaches seem to overcome modularity-optimization based approaches for community detection. Further experiments are needed to explore the performances gain when applying the approach using more sophisticated basic graph-based tag recommenders. We target mainly applying the coarsening approach using a link prediction based tag recommender [8].

References

- [1] Gupta, M., Li, R., Yin, Z., Han, J.: Survey on social tagging techniques. *SIGKDD Explorations* **12**(1) (2010) 58–72
- [2] Milicevic, A.K., Nanopoulos, A., Ivanovic, M.: Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions. *Artif. Intell. Rev.* **33**(3) (2010) 187–209
- [3] Fang, X., Pan, R., Cao, G., He, X., Dai, W.: Personalized tag recommendation through nonlinear tensor factorization using gaussian kernel. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, January 25-30, 2015, Austin, Texas, USA. (2015) 439–445

- [4] Feng, W., Wang, J.: Incorporating heterogeneous information for personalized tag recommendation in social tagging systems. In Yang, Q., Agarwal, D., Pei, J., eds.: The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012, ACM (2012) 1276–1284
- [5] Gemmell, J., Mobasher, B., Burke, R.D.: User partitioning hybrid for tag recommendation. In: User Modeling, Adaptation, and Personalization - 22nd International Conference, UMAP 2014, Aalborg, Denmark, July 7-11, 2014. Proceedings. (2014) 74–85
- [6] Jäschke, R., Marinho, L.B., Hotho, A., Schmidt-Thieme, L., Stumme, G.: Tag recommendations in folksonomies. In: Knowledge Discovery in Databases: PKDD 2007, Warsaw, Poland (2007) 506–514
- [7] Papadopoulos, S., Kompatsiaris, Y., Vakali, A.: A graph-based clustering scheme for identifying related tags in folksonomies. In: DaWak. (2010) 65–76
- [8] Pujari, M., Kanawati, R.: Tag recommendation by link prediction based on supervised machine learning. In: Sixth International AAAI Conference on Weblogs and Social Media (ICWSM'2012), Dublin (June 2012) 547–550
- [9] Rawashdeh, M., Alhamid, M.F., Kim, H., Alnusair, A., Maclsaac, V., El-Saddik, A.: Graph-based personalized recommendation in social tagging systems. In: 2013 IEEE International Conference on Multimedia and Expo Workshops, Chengdu, China, July 14-18, 2014. (2014) 1–6
- [10] Rawashdeh, M., Kim, H., Alja'am, J.M., El-Saddik, A.: Folksonomy link prediction based on a tripartite graph for tag recommendation. *J. Intell. Inf. Syst.* **40**(2) (2013) 307–325
- [11] Gueye, M., Abdessalem, T., Naacke, H.: Strec: An improved graph-based tag recommender. In: Proceedings of the Fifth ACM RecSys Workshop on Recommender Systems and the Social Web co-located with the 7th ACM Conference on Recommender Systems (RecSys 2013), Hong Kong, China, October 13, 2013. (2013)
- [12] Kim, H.N., El-Saddik, A.: Personalized pagerank vectors for tag recommendations: inside folkrank. In: ACM conference on Recommender systems. (2011) 45–52
- [13] Guan, Z., Bu, J., Mei, Q., Chen, C., Wang, C.: Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009. (2009) 540–547
- [14] Song, Y., Zhang, L., Giles, C.L.: Automatic tag recommendation algorithms for social recommender systems. *TWEB* **5**(1) (2011) 4
- [15] Kubatz, M., Gedikli, F., Jannach, D.: Localrank - neighborhood-based, fast computation of tag recommendations. In: E-Commerce and Web Technologies - 12th International Conference, EC-Web 2011, Toulouse, France, August 30 - September 1, 2011. Proceedings. (2011) 258–269
- [16] Zhang, Z.K., Zhou, T., Zhang, Y.C.: Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs. *CoRR* **abs/0904.1989** (2009)
- [17] Liu, Z., Chi, C., Sun, M.: Folkdiffusion: A graph-based tag suggestion method for folksonomies. In: Information Retrieval Technology, Springer Berlin / Heidelberg (2010) 231–240
- [18] Kanawati, R.: Multiplex network mining: a brief survey. *IEEE Intelligent Informatics Bulletin* **16** (2015) 24–28
- [19] Hmimida, M., Kanawati, R.: Community detection in multiplex networks: a seed-centric approach. *Networks and Heterogeneous Media* **10**(1) (March 2015) 71–85
- [20] Fortunato, S.: Community detection in graphs. *Physics Reports* **486**(3-5) (2010) 75–174
- [21] Berlingerio, M., Pinelli, F., Calabrese, F.: Abacus: frequent pattern mining-based community discovery in multidimensional networks. *Data Min. Knowl. Discov.* **27**(3) (2013) 294–320
- [22] Cai, D., Shao, Z., He, X., Yan, X., Han, J.: Mining hidden community in heterogeneous social networks. In: ACM-SIGKDD Workshop on Link Discovery: Issues, Approaches and Applications (LinkKDD'05), Chicago, IL (Aug 2005)
- [23] Berlingerio, M., Coscia, M., Giannotti, F.: Finding and characterizing communities in multidimensional networks. In: ASONAM, IEEE Computer Society (2011) 490–494

- [24] Strehl, A., Ghosh, J.: Cluster ensembles: a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research* **3** (2003) 583–617
- [25] Goder, A., Filkov, V.: Consensus clustering algorithms: Comparison and refinement. In Munro, J.I., Wagner, D., eds.: *ALLENEX, SIAM* (2008) 109–117
- [26] Topchy, A.P., Jain, A.K., Punch, W.F.: Clustering ensembles: Models of consensus and weak partitions. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(12) (2005) 1866–1881
- [27] Kanawati, R.: Empirical evaluation of applying ensemble methods to ego-centered community identification in complex networks. *Neurocomputing* **150, B** (February 2015) 417–427
- [28] Mucha, P.J., Richardson, T., Macon, K., Porter, M.A., Onnela, J.P.: Community structure in time-dependent, multiscale, and multiplex networks. *Science* **328**(5980) (2010) 876–878
- [29] Reichardt, J., Bornholdt, S.: Statistical mechanics of community detection. *Physical Review E* **74**(1) (2006)
- [30] Blondel, V.D., Guillaume, J.L., Lefebvre, E.: Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* **2008** (2008) P10008
- [31] Pons, P., Latapy, M.: Computing communities in large networks using random walks. *J. Graph Algorithms Appl.* **10**(2) (2006) 191–218
- [32] Rosvall, M., Axelsson, D., Bergstrom, C.T.: The map equation. *Eur. Phys. J. Special Topics* **13** (2009) 178
- [33] Kuncheva, Z., Montana, G.: Community detection in multiplex networks using locally adaptive random walks. In: *MANEM 2workshop - Proceedings of ASONAM 2015, Paris* (August 2015)
- [34] Domenico, M.D., Lancichinetti, A., Arenas, A., Rosvall, M.: Identifying modular flows on multilayer networks reveals highly overlapping organization in social systems. *Phys. Rev* **5** (2015) 011027
- [35] Yakoubi, Z., Kanawati, R.: Licod: Leader-driven approaches for community detection. *Vietnam Journal of Computer Science* **1**(4) (2014) 241–256
- [36] Kanawati, R.: Seed-centric approaches for community detection in complex networks. In Meiselwitz, G., ed.: *6th international conference on Social Computing and Social Media. Volume LNCS 8531.*, Crete, Greece, Springer (June 2014) 197–208
- [37] Pujari, M., Kanawati, R.: Applying supervised rank aggregation to link prediction in large scale complex networks. In: *Journée : Big data mining and visualization, Tours* (June 2012)
- [38] Battiston, F., Nicosia, V., Latora, V.: Metrics for the analysis of multiplex networks. *CoRR abs/1308.3182* (2013)
- [39] Cantador, I., Brusilovsky, P., Kuflik, T., eds.: *2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011).*, York, NY, USA, ACM (2011)
- [40] Falih, I., Kanawati, R.: Muna: A multiplex network analysis library. In: *The 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Paris* (August 2015) 757–760

Community detection in visibility networks: an approach to categorize percussive influence on audio musical signals

Dirceu de Freitas Piedade Melo, Inacio de Sousa Fadigas and Hernane Borges de Barros Pereira

Abstract The feature extraction is a very important step in the music audio classification. This task has been performed by renowned descriptors using, in most cases, the time-frequency approach. In this article we propose a descriptor that performs the feature extraction in a set of music audio files labeled in symphonic and percussive music, using parameters calculated within the Euclidean domain. First we calculate the variance fluctuation series of music signal, after we map this series into visibility graphs [13]. At the end each audio track will correspond to a network, where the links are defined by the visibility of variance fluctuations of their respective audio signal. Then, we measure the strength of the partitions of each network in clusters, using calculation of modularity. The results of computation of this parameter in sixty networks showed that percussive and symphonic music can be distinguished and hierarchized on a growing rang, following a direct correlation with modularity.

1 Introduction

Due to the need to develop computational resources for the organization of large digital music libraries, the importance of automatic music classification systems has grown considerably in recent times [19]. Many classification platforms have been proposed [6, 8, 10, 20], and despite efforts to find a new path [9, 12], most feature extraction tools use knowledge of the audio signal processing field [2, 7, 23, 26]. Among the most commonly descriptors used in feature extraction are MFCC - Mel

Dirceu de Freitas Piedade Melo (e-mail: dirceumelo@ymail.com)✉

Department of Mathematics (DEMAT), Nucleus of Studies of Mathematics, Statistics and Education (NEMEE), Federal Institute of Education Science and Technology of Bahia (IFBA), Brazil

Inácio de Souza Fadigas (e-mail: isfadigas@gmail.com)

State University of Feira de Santana (UEFS), Bahia, Brazil

Hernane Borges de Barros Pereira (e-mail: hbbpereira@gmail.com)

State University of Bahia (UNEB), Computational Modeling Program, SENAI CIMATEC, Bahia, Brazil

Frequency Cepstral Coefficients, Spectral Rollof, Spectral Flux, Zero Crossing Rate, Low-Energy Feature. These algorithms lead their mathematical operations in time-frequency domain in order to extract of the musical signal, three basic characteristics: tone texture (timbre), rhythmic content (time, rhythm, pulse), and tonal content (pitch) [2]. Hoping to cooperate for the growth of new ways to perform feature extraction in musical audio signals, we propose in this article a way to describe musical dynamics¹ in audio tracks following a different direction. To make possible this idea we first captured the loudness of the audio signal from the calculation of the average intensity of their fluctuations in fixed-size windows [12], creating a series of variance fluctuations of the original signal. After this, we mapped this series into a graph, using the geometrical visibility mapping proposed by [13]. In this mapping, if two points of the series see each other in the Cartesian plane, an edge is created in the Euclidean plane. Thus the higher the visibility of a point in the series, the more edges it will have in the graph. At the end of the mapping, the graph inherited in its structure the visibility of all local peaks with their respective neighborhood [15]. Consequently, variance fluctuation series with few local peaks, but very visible, will generate graphs with few hubs, but with a high degree of connections. On the other hand, series with many local peaks with poor visibility will generate graphs with many vertices with lower level of connections. The analysis of modularity will identify if the network structure was created from the series with greater or smaller local visibility. The experiments suggest that the visibility graph generated from the variance fluctuations of audio signals that have a strong influence of percussion activity - like Samba or disco music- have a higher trend to create modules than audio signals whose orchestration has little or no influence of percussion instruments and more dynamics nuances, like a string quartet.

2 Related Works

Researchers at the computer music area have used the structural feature of complex networks to solve various problems related to music information retrieval, such as: musical taste in internet communities [4], algorithmic composition [25], collaborative networks between composers [21], music genre classification [5]. In [25] authors build a network based on pattern analysis of Bach, Chopin and Mozart compositions, linking the duration of two notes in MIDI (Musical Instrument Digital Interface) format which co-occur in a melodic phrase, using universal properties found in these networks to propose rules for algorithmic composition. To analyze the musical tastes of users from their playlists, [4] uses the basic features of networks where the nodes are the song titles, and the edges occur between two song titles, if this title appear in more than one playlist.[5] deals with music genre classification using rhythms extracted from MIDI database, transforming it into complex networks. In [5] each rhythmic cell is a node, while the sequences of notes define the links between nodes,

¹ The varying levels of volume of sound in different parts of a musical performance.<https://en.oxforddictionaries.com/definition/dynamics>.

according to a Markov model. [11] combines audio analysis and network structures to identify communities of artists on myspace website, establishing links between two artists who have similar tags on social networks, and audio-based similarity using Mel Frequency Cepstral Coefficients, and entropy. [21] studies the topology and evolution of networks of western classical music composers, building links between two composers who co-occur in the same compact disc, linking information about author, period and style extracted from audio file meta-data. A characteristic that can be noticed in most scientific papers that use the mapping of complex networks to understand the music audio phenomena is the absence of structures formed by links where the nodes are non-symbolic elements. With the exception of [11], which use audio data in the network vertex in the first of two phases of the mapping, we have not found in the survey of related work, another study whose network is formed by the relationship between audio signal points. Considering the survey by [22], that shows various approaches for music content analysis, we also note the lack of methodologies that use complex network parameters to perform feature extraction in audio signals.

Visibility graphs have been created bridges between time series analysis and complex networks analysis, opening possibilities on time series field by using a set of new tools. One of this bridges has been used to study long-term correlations, fractal properties, and self-similarity structures [14, 18] and have found applications in temporal observations like Nasdaq and *S&P500* daily stock indices [24] and traffic of information packets series [1]. These studies show that the visibility graphs has the ability to capture local trends of time series and measure them through the network analysis. Motivated by these studies, we chose the same type of mapping seeking to identify how much the persistence of an audio signal time series is associated with the dynamics changes influenced by percussive activity of its musical content. This article will show that modularity is able to capture the reflections of the self-similarity and patterns of persistence of loudness embedded in the network, but will not establish a direct relationship with power laws or the Hurst exponent calculations, as in [14].

3 Materials and Methods

In this section we first present the database, after we show the methodological approach to conduct the study of the visibility of an audio signal by using the modularity of complex networks. We take a set of sixty audio samples with 30 seconds long. Each song is represented by a time series $W(i)$. In this series we calculate the subset of variance fluctuations $V(j)$. For each $V(j)$ point is evaluated the "visibility" in relation to their successors and predecessors, according to the slope comparisons [13]. At the end of the process the subset $V(j)$ becomes the graph $G(V(n), V(m))$, from which is estimated the modularity and the amount of communities.

3.1 Database

The audio files used In this article are divided into two groups: Symphonic Music and Percussive Music. In Symphonic Music were selected thirty compositions for string quartet or large orchestra. The compositions are divided among Bach concertos, Mozart symphonies and string quartets by Debussy and Ravel. To represent the Percussive Music, we chose 30 tracks equally divided into: samba, mangue beat and disco music. The Samba tracks are songs composed for the celebration of the Rio de Janeiro carnival from 2005 to 2014. In Mangue Beat there is an influence of electronic pop-rock music, mixed with a traditional afro-brazilian rhythm called Maracatú. The ten tracks of disco music gives a good overview of the musical scene of the 80s. The symphonic and disco music are chosen from GTZAN ² database, and samba e mangue beat are from the author's particular collection. The Percussive tracks are labeled as Percussive 1 ... Percussive 30, where disco music occupies the ten first places, samba takes up the next ten, and Mangue Beat the past ten. The Symphonic networks are labeled as Symphonic 1 ... Symphonic 30, where the eight first are Bach concertos, the next sixteen networks are Mozart Symphonies, and the last six are string quartets composed by Debussy, Dutilleux and Ravel.

3.2 Calculating The Variance Fluctuation Series

In this section we first calculate the variance fluctuations of a musical signal with the same methodology used by [12, 16], where the authors consider that the loudness can be represented by average intensity of the sound over intervals of 0.01 s. Consider audio music signal represented by the $W(i)$ series, with $i = 1 \dots N$. The total number of points N is a function $N = SR.t$, where the sampling rate is $SR = 11kHz$ and the time is $t = 30$ seconds. The set $W(i) = W(1), \dots, W(N)$, with $N = 330,000$ is segmented into m -non overlapping boxes $\lambda = 110$. For each box $j = 1 \dots m$ is calculated by the standard deviation. In j^{th} box we have:

$$V(j) = \sqrt{\frac{\sum_{(j-1)\dots\lambda+1}^{j\lambda} (W(i) - \bar{W}(j))^2}{\lambda - 1}}, \quad (1)$$

Where the average is given by:

$$\bar{W}_j = \frac{\sum_{(j-1)\dots\lambda+1}^{j\lambda} (W(i))}{\lambda} \quad (2)$$

This creates the variance fluctuation subseries $V(j) = V1, V2, \dots, Vm$, with 3000 samples.

² Gtzan Genre Collection is a database widely used in musical information retrieval research. It was proposed by 8 and is available at http://marsyasweb.appspot.com/download/data_sets

3.3 Transforming Variance Fluctuations in Graphs

Each variance fluctuation point $V(j)$, with $j = 1 \dots 3000$, is considered to be a vertex of the network. To apply the visibility criterion in the series, we will consider each point of $V(j)$ as an ordered pair (x_j, V_j) , where x_j is the point position in the series. Two vertex (x_a, V_a) and (x_b, V_b) are connected if there is a point (x_c, V_c) between the m such that:

$$\frac{V_b - V_c}{x_b - x_c} > \frac{V_b - V_a}{x_b - x_a} \quad (3)$$

Equation 3 proposed for [13], provides a comparison between the α_{bc} slope (left side of equation) and α_{ba} slope (right side of equation). Whenever $\alpha_{bc} > \alpha_{ba}$ there is visibility between V_a and V_b , and their corresponding nodes are connected in the graph. Otherwise, they do not constitute an edge in the graph. After the equation 3 is applied to all points of the series, following the order $j = 1 \dots 300$, we have the visibility of each point of a subset $V(j)$ mapped in a graph $(V(m), V(n))$. This means that, from this stage, each song is represented by a visibility graph.

3.4 Modularity

After mapping the variance fluctuations into visibility graph, the modularity is calculated using the Lovain Method [3], based on GEPHI³ framework for community detection. This algorithm brings a fast unfolding approach for the fundamental modularity defined for [17], whose equation is

$$Q = \frac{1}{2m} \sum_{(i,j)} \left(A_{ij} - \frac{k_i - k_j}{2m} \right) \delta(c_i, c_j) \quad (4)$$

Where i and j are nodes of the network; A_{ij} represents the number of edges between i and j ; k_i and k_j are the sum of the the edges attached to i and j ; m is the sum of all edges in the graph; c_i and c_j are the communities of the nodes; and $\delta(c_i, c_j)$ is a Kronecker delta function 0 for $c_i = c_j$ and 1 for $c_i \neq c_j$; where c_i and c_j are the communities of the nodes.

To maximize the modularity efficiently, Louvain method proposes a method which uses two stages in iterative repetitions: (1) each node is attributed to their own community. So the change of modularity is calculated for each node i , removing this node from its own community C and moving it to the community of each neighbor i . This value can be easily calculated by:

$$\Delta Q = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m} \right)^2 \right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right] \quad (5)$$

³ GEPHI is a free and open-source software that performs visualization and operation of all types of graphs and networks. Available in <https://gephi.org/>.

Where \sum_{in} is the sum of the links inside C ; \sum_{tot} is the sum of the links incident to nodes in C ; $k_{i,in}$ is the sum of the links incident to node i ; m is the sum of the links from i to nodes in C and m is the sum of the weights of all the links in the network. In the second stage the nodes belonging to the same community are united, and then it constructed a new network where the nodes are small communities. These steps are repeated until the maximum modularity is achieved and a community hierarchy is produced.

Since the calculation of modularity depends on a random argument, the algorithm each time will return different results. With the tested networks there was very little variation in these results, therefore we considered to all networks a randomly selected result.

4 Experimental results

4.1 Variance fluctuations

Sixty variance fluctuation series were calculated by reducing the original signal, approximately 330,000 points for 3000 points. Figure 1 illustrates the variance fluctuation series of two audio samples. The first represents the Percussive group and the second the Symphonic group. In Figure 1(a) we have a numerical series generated from a song with a strong beat of drums, used in traditional Brazilian rhythm "maracatu", and Figure 1(b) the portion of a Mozart symphony performed by string section of an orchestra, without percussion instruments.

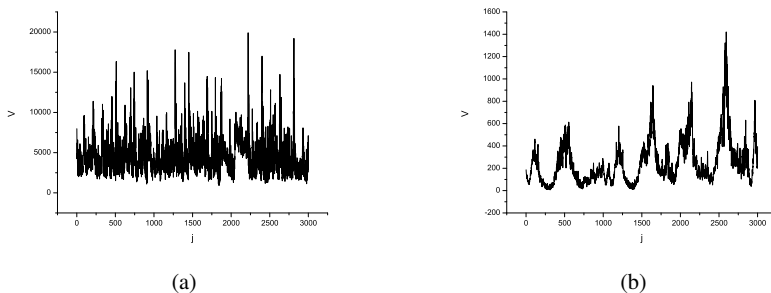


Fig. 1: Variance Fluctuations series of audio signals corresponding to the files: (a) Etnia by Chico Science & Zoombie Nation; (b) Mozart -Symphony 39 in E flat Major, K 543. Source: Author

We can notice by visual inspection that the first series is denser than the second, with less space between "peaks" and "valleys". We can, even without numerical proof, intuit that these different geometric configurations are associated with the

peculiar rhythmic activities to their audio signals. The following results present quantitative basis for characterizing these differences.

4.2 Visibility networks generated from variance fluctuation

We mapped Sixty networks, each with 3000 nodes. The networks are grouped into two types: Symphonic Networks and Percussive Networks. Figure 2 shows two networks, representing respectively the Percussive and Symphonic Networks. The first (Figure 2a) is a mapping of a audio from the 1980s - So Many Men, so Little Time - played by the Canadian singer Miquel Brown. The second (Figure 2b) is a network generated by the mapping of the audio Animé Et Très Décidé - String quartet composed by Claude Debussi, performed by Julliard String Quartet. In these two representations, the modularity classes appear in different colors, indicating the communities formed by each network. Sections 4.3 and 4.4 will present overall results that will give subsidy to infer about trends presented by each group, based on the magnitude of the difference between the amounts of communities formed by the two types of networks.

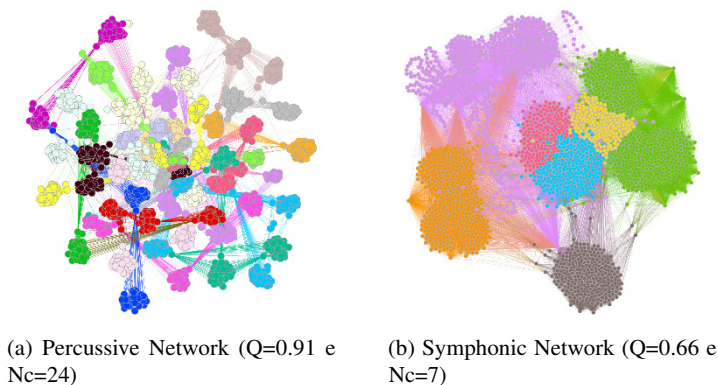


Fig. 2: Visibility Networks of the variance fluctuations of two audio signals. The colors represent the modularity classes of each network. Source: Author.

The average number of edges of 30 Symphonic and Percussive Networks are, respectively, 60254 ± 10925 and 23827 ± 2899 . The results show a significant difference between the mean values of edges generated between the two types of networks. Taking into account that the number of nodes in visibility graphs depends on the visibility of their points in the series. We can infer that, in mapping a set, the greater the number of nodes generated, the higher the visibility of the series. This indicates that, on average, the series that generated the Symphonic networks have greater visibility than the generating series of Percussive Networks.

4.3 Modularity

The results of modularity (Q) of the 60 visibility networks are shown in Figure 3. The networks of each group are indicated with the numbers 1 to 30. We note that all the Q values for Percussive Network ($\langle Q \rangle = 0.81 \pm 0.08$) are higher than the values calculated for Symphonic Network ($\langle Q \rangle = 0.54 \pm 0.13$). The extreme values of modularity are 0.91 for the visibility network of the song Get Up played by the british african-pop band Osibisa, and 0.14 for the network of the Symphony 39 in E flat Major - k 543 composed by Mozart. The Symphonic Networks showed a set of less compact modularity values in the average, with a 12% deviation against the 8% of the Percussive Networks, even so, the average of the two groups showed significant differences with a confidence of 95%, according to the Bonholm test. At this point we can infer, based on the arguments presented in section 4.2, and also on the Q values calculated, that there is an inverse relationship between visibility and modularity.

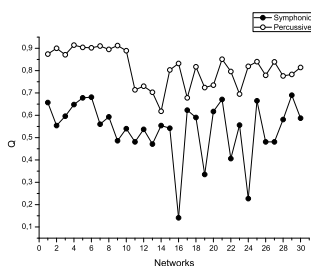


Fig. 3: Modularity of 30 Symphonic (black dots) and Percussive (white dots) Visibility Networks. Source: author.

4.4 Number of communities

Each Q value calculated in section 4.3 is associated with a number of communities (N_c) of the network. Figure 4 shows the N_c values calculated for each Q (Figure 3). Globally the amount of network communities follow the same feature found in the calculation of modularity: exists a very clear distinction between the two classes, where the Percussive Networks outweigh the Symphonic networks for most N_c values. The average values obtained were $\langle N_c \rangle = 16.5 \pm 4.4$ for Percussive, and $\langle N_c \rangle = 8.8 \pm 2.2$ for Symphonic networks. Looking locally we can see that in addition to the distinction into two groups, N_c values of Percussive Network can serve as a parameter for stratification within the group, in order, for example, the great distance of the first nine networks N_c values (white dots) to the rest.

4.5 Influence of the randomness in the results

In this section we present the results of a study made about the influence of the randomness factor in the calculation of modularity. As discussed in section 3.4, the calculation of modularity is made based on the comparison of information given for

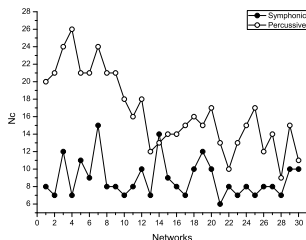


Fig. 4: Number of communities (Nc) of 30 Symphonic (black dots) and Percussive (white dots) of Visibility Networks of audio signal variance fluctuations. Source: author.

edges that exist on the network and edges made randomly. Each time the algorithm is applied, we obtain a value for the modularity and the number of communities. Table 2 shows ten takes from the calculation of modularity and the number of communities for one Percussive Network. In this table we can see that in some cases, the algorithm estimates the same modularity for different Nc values (Takes 2 and 3), and the same number of communities for different Q values (Takes 5 and 9). This shows that, due to the random factor, there is no modularity value associated with a unique modularity class arrangement. In Table 1 we have $\langle Q \rangle = 0.8420 \pm 0.0022$ and $\langle Nc \rangle = 15.90 \pm 0.99$ for ten takes. Increasing the number of repetitions to 80 takes we have $\langle Q \rangle = 0.8422 \pm 0.0024$ and $\langle Nc \rangle = 16.15 \pm 0.80$. Comparing the results obtained for the two tests, it is clear that the means and variances of Q and Nc do not change significantly with increasing the number of takes, and that for this network there is a great chance that if we choose one of ten or eighty attempts, we find a value of Q and Nc very close to the same average value.

Now we will show the results that investigate the overall impact of the random factor in the calculation of the modularity. We calculate ten repetitions of the Nc of twenty networks (Figure 5). In the x-axis, the networks S1 to S10 (white boxes) are Symphonic Networks, and networks P1 to P10 (dashed boxes) are Percussive Networks. We can see that the overall behavior does not change with the recalculation for each network.

4.6 About sample rate changes

In order to investigate the impact of sample rate changes in results of network parameters we calculate average degree ($\langle k \rangle$), density (Δ), modularity (Q), number of communities (Nc), diameter (D), average path length (L), clustering coefficient (C), and time processing (TP), of the visibility networks Percussive 11 (samba) and Symphonic 1 (oboé concert), using three sampling rates (SR): 11025 Hz, 22050 Hz and 44100 Hz. We use the framework Gephi 0.9.0 to calculate all parameters.

The results in Table 2 show that SR changes do not bring significant differences in the final statistics, neither alter the trends found in the comparative study between the two musical groups. The computational processing time recorded for this experiment

Table 1: Calculation of the modularity of the network "A walk in the free world" written by Chico Science and the Zombie Nation with ten repetitions. (Source: author).

Take	Modularity (Q)	Communities (Nc)
1	0.844	16
2	0.844	17
3	0.844	16
4	0.839	17
5	0.839	15
6	0.843	16
7	0.842	17
8	0.845	14
9	0.843	15
10	0.844	16

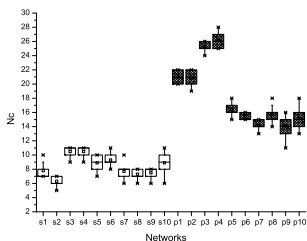


Fig. 5: Number of communities (Nc) of ten Symphonic Visibility Networks (white boxes) and ten Percussive Visibility Networks (dashed boxes), with ten calculations each. (Source: author).

had the decisive influence of the diameter and the average path length calculation. To process only these two parameters, the Gephi spent more than 90% of the total time. If the calculation of these parameters for many networks is required, the rate of 22 and 44kHz are not recommended. To calculate only Q and Nc the Gephi took around 1 sec for each SR.

4.7 Looking closely at some Percussive and Symphonic Networks

Observing the Figure 3 we can see that some points stood out from the rest of the group because they have reached discrepant or extreme values. Below we will discuss the possible causes of this behavior, putting together musical and statistics similarities.

- *Networks P1 to P10* - They achieved greater magnitude and shorter variance in modularity ($\langle Q \rangle = 0.897 \pm 0.015$) compared to P11-P30 ($\langle Q \rangle = 0.767 \pm$

Table 2: Network parameters of networks Percussive 11 (P11) and Symphonic 1 (S1) for 3 different sample rates used in his respective audio samples before network mapping.

Network	SR (hz)	V	E	$\langle k \rangle$	Δ	Q	Nc	D	L	C	TP (sec)
P11	11025	3000	18866	12.58	0.004	0.857	17	7	3.85	0.837	23
	22050	6000	41500	13.85	0.002	0.875	19	10	3.745	0.845	160
	44100	12000	83179	13.87	0.001	0.918	27	9	4.296	0.849	734
S1	11025	3000	65115	44.74	0.015	0.657	8	4	2	0.860	90
	22050	6000	152262	50.754	0.008	0.714	10	4	1.998	0.878	61
	44100	12000	299272	49.895	0.004	0.682	8	5	2	0.902	3180

0.064)) and Symphonic networks (Section 4.3). Looking at the distributions of vertices per community, of all networks, we observed higher homogeneity in P1-P10 distributions. This contributed to these networks have obtained greater modularity than the others. Fig 6 shows the distribution of vertices per community of P6, representing P1-P10 networks, and P27, representing the others percussive networks. Comparing the two distributions we can notice greater homogeneity in P6, which reached modularity 0.902, while P27, with less homogeneity got $Q=0.839$. Musically the P1-P10 networks represent songs of the eighties, which is characterized by danceable groove on every song, dominated by the constant pulse of bass and drums without much dynamics variation. We can speculate that this "musical homogeneity" may have strongly influenced the statistical uniqueness that made these networks stood out from all others.

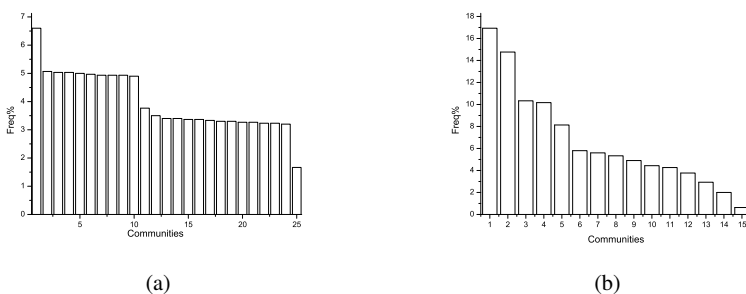


Fig. 6: Distribution of vertices per community of the networks:(a) Percussion 4 - Disco Music, and (b) Percussion 27 - Mangue Beat. Source: Author.

- *Networks S16, S19, S24* - These networks draw attention by having modularity with very low values (0.141, 0.227 and 0.355). Musically, the audio excerpts associated with these networks also have a common feature. In all of them there

is a sudden change of dynamics, strongly influenced by the presence or absence of timpani⁴. It created a particular topology in the variance fluctuations of these audio signals, with great "valleys" followed by high "peaks", favoring visibility graphs with big hubs, and cluster distributions with very low amount of nodes in some communities. In consequence, they achieved lower modularity values than the others symphonic networks. Figure 7 (a) shows the variance fluctuations of the audio track Symphonic 16. We can note in Figure 7 (b) that five communities have less than 5% of vertices, while only one community have about 50% of them. The modularity maximization algorithm was not able to merge these small communities into larger communities. This prevented the Q value to stay a bit higher. Anyway the lower Q values found in these three networks, helped to distinguish the particular audio musical behavior that these networks are topologically representing.

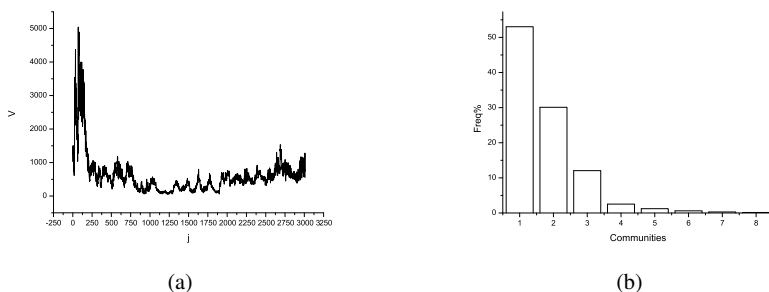


Fig. 7: (a) Variance fluctuations of the track S16 - Symphony 39 in E flat Major, K 543, Mozart. During the first two seconds ($j = 200$ to 3000), the whole orchestra, including timpani, play a part in fortissimo, and thereafter comes off the timpani, and remain the strings and woods gently touching; (b) Relative frequency of vertices by community of the S16 visibility network. Source: Author.

5 Conclusion and future work

In this article we mapped variance fluctuations of sixty musical audio files into visibility graphs, and through the modularity and the number of communities of each network, we measured the level of dynamics changes influenced by percussive activity of each audio content. We concluded that modularity and number of communities of complex networks has produced useful information for categorization

⁴ A set of two or three large drums (called kettledrums) that are played by one performer in an orchestra <http://www.merriam-webster.com/dictionary/timpani>.

into two groups, where audio samples with musical affinities were gathered within the same group according to its high or low percussive activity. Although in this study we have explored the feature extraction with only two categories, the algorithm showed potential for categorizing by more than two labels. Other investigations are in progress in which some network features are performing an audio music hierarchy according to the taxonomy of some musical genres, with a large number of files. To better understand the level of contribution that this algorithm can give to the music information retrieval field, we will conduct an experiment comparing the parameters extracted from the variance visibility networks with rhythm-based tools most used in the literature. Another important issue which is worth be discussed in future work is the evaluation of Pajek adjustment indices (Cramer's V, Rajsiki and Adjusted Rand Index) in front of the parameters adopted by Gephi, and its influence on the extraction of features proposed by the visibility descriptor of variance fluctuations.

References

- [1] Andjelković, M., Gupte, N., Tadić, B.: Hidden geometry of traffic jamming. *Physical Review E* **91**(5), 052,817 (2015)
- [2] Bergstra, J., Casagrande, N., Eck, D.: Two algorithms for timbre and rhythm-based multiresolution audio classification. In: *Proceedings of ISMIR* (2005)
- [3] Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* **2008**(10), P10,008 (2008)
- [4] Buldú, J.M., Cano, P., Koppenberger, M., Almendral, J.A., Boccaletti, S.: The complex network of musical tastes. *New Journal of Physics* **9**(6), 172 (2007)
- [5] Correa, D.C., Saito, J.H., da F Costa, L.: Musical genres: beating to the rhythms of different drums. *New Journal of Physics* **12**(5), 053,030 (2010)
- [6] Costa, Y.M., Oliveira, L., Koerich, A.L., Gouyon, F., Martins, J.: Music genre classification using lbp textural features. *Signal Processing* **92**(11), 2723–2737 (2012)
- [7] Eronen, A.: *Signal processing methods for audio classification and music content analysis*. Tampereen teknillinen yliopisto. Julkaisu-Tampere University of Technology. Publication; 817 (2009)
- [8] Ezzaidi, H., Rouat, J.: Automatic musical genre classification using divergence and average information measures. *World Academy of Science, Engineering and Technology* **15** (2006)
- [9] Goulart, A.J.H.: *Classificação automática de gênero musical baseada em entropia e fractais*. Ph.D. thesis, Universidade de São Paulo
- [10] Gaus, E., et al.: *Audio content processing for automatic music genre classification: descriptors, databases, and classifiers* (2009)
- [11] Jacobson, K., Sandler, M.B., Fields, B.: Using audio analysis and network structure to identify communities in on-line social networks of artists. In: *ISMIR*, pp. 269–274 (2008)
- [12] Jennings, H.D., Ivanov, P.C., Martins, A.d.M., da Silva, P., Viswanathan, G.: Variance fluctuations in nonstationary time series: a comparative study of music genres. *Physica A: Statistical Mechanics and its Applications* **336**(3), 585–594 (2004)
- [13] Lacasa, L., Luque, B., Ballesteros, F., Luque, J., Nuno, J.C.: From time series to complex networks: The visibility graph. *Proceedings of the National Academy of Sciences* **105**(13), 4972–4975 (2008)

- [14] Lacasa, L., Luque, B., Luque, J., Nuno, J.C.: The visibility graph: A new method for estimating the hurst exponent of fractional brownian motion. *EPL (Europhysics Letters)* **86**(3), 30,001 (2009)
- [15] Lacasa, L., Toral, R.: Description of stochastic and chaotic series using visibility graphs. *Physical Review E* **82**(3), 036,120 (2010)
- [16] Melo, D.F.P.: Análise de flutuações de variância em sinais de áudio agrupados por gênero musical. *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics* **1**(1) (2013)
- [17] Newman, M.E.: Analysis of weighted networks. *Physical review E* **70**(5), 056,131 (2004)
- [18] Nunez, A., Lacasa, L., Valero, E., Gómez, J.P., Luque, B.: Detecting series periodicity with horizontal visibility graphs. *International Journal of Bifurcation and Chaos* **22**(07), 1250,160 (2012)
- [19] Pampalk, E., Rauber, A., Merkl, D.: Content-based organization and visualization of music archives. In: *Proceedings of the tenth ACM international conference on Multimedia*, pp. 570–579. ACM (2002)
- [20] Panagakis, Y., Kotropoulos, C., Arce, G.R.: Music genre classification via sparse representations of auditory temporal modulations. In: *Signal Processing Conference, 2009 17th European*, pp. 1–5. IEEE (2009)
- [21] Park, D., Bae, A., Schich, M., Park, J.: Topology and evolution of the network of western classical music composers. *EPJ Data Science* **4**(1), 1 (2015)
- [22] Schedl, M., Gómez, E., Urbano, J., et al.: *Music information retrieval: Recent developments and applications*. Now Publ. (2014)
- [23] Silla Jr, C.N., Kaestner, C.A., Koerich, A.L.: Automatic genre classification of latin music using ensemble of classifiers. In: *Proc. of the 33rd Integrated Software and Hardware Seminar*, pp. 47–53 (2006)
- [24] Stephen, M., Gu, C., Yang, H.: Visibility graph based time series analysis. *PloS one* **10**(11), e0143,015 (2015)
- [25] Tse, C., Liu, X., Small, M.: *Analyzing and composing music with complex networks: finding structures in bach, chopin and mozart* (2008)
- [26] Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing* **10**(5), 293–302 (2002)

Can we recognize the next user's mobile community?

Ahlem Drif, Abdellah Boukerram, Yacine Slimani and Silvia Giordano

Abstract Accurate location prediction is central for the current and future location based services. We propose here an approach based on a new definition of community, which is centered on individual interests, and open for a novel prediction approach that exploits the properties of these communities. We show on real traces that the proposed approach is very efficient and allows to achieve high performances.

1 Introduction

Predicting individual's next movements using his/her past history and also the history of people related to him/her is one of the most interesting research areas in computational social science. Wang et al [11] have studied and analyzed the trajectories and communication records of 6 Million mobile phone users. The authors have proved, by combining the measurements of network proximity and mobile homophily, that the similarity between two individuals movements is strongly correlated with their proximity in the social network. In [10], Pang et al have determined the check-in geographic regions and identified communities of user's friend on the tweeter network, and have demonstrated that communities' influences on users' mobility are stronger than their friends' and each user is only influenced by a small number of his/her communities. Garg and al [5] proposed a new prediction algorithm based on users interest profile and the mobility history of the community. They have illustrated

Ahlem Drif (e-mail: adrif@univ-setif.dz)✉
Networks and Distributed System Laboratory, University of Setif 1, Algeria.

Abdellah Boukerram (e-mail: boukerram@hotmail.com)
Computer Science Department, University of Bejaia, Algeria.

Yacine Slimani (e-mail: slimani_y09@univ-setif.dz)✉
Laboratory of Intelligent Systems, University of Setif 1, Algeria.

Silvia Giordano (e-mail: silvia.giordano@supsi.ch)
Networking Lab, University of Applied Sciences of Southern Switzerland, SUPSI, Switzerland.

that a single user in his/her own visiting behaviour tends to be more conservative than looking at himself within a crowd of people and the overall community tends to deviate from its regularity more easily than a single user.

In our previous work [3], we have identified Interest Based Mobile Communities, called IMoComm, for mobile users. Interest that seems to be the main reason that motivates individuals to move from one place to another. In fact, the extraction of a user's sequence of activities and the share of interests with some other users allows to predict the likelihood that the user will behave in a particular way and to define the probability of choosing a location close to his/her group of interest. In this paper, we aim to improve such prediction by exploiting additional available information included in the IMoComm. Intuitively, an individual tends to join a community of his/her interests that is varying over time but his/her move is strongly connected to his/her social preferences, career goals, and daily life habits. Thereby, the extraction of community link patterns helps predicting his/her future movement by incorporating useful information conveyed by users communities ties while tracking his/her mobility history. The link prediction problem has attracted immense interest in recent years, and a variety of techniques that operate on the graph/hypergraph structure of social networks are proposed. For a full review of the state of the art in link prediction in social networks, see Peng et al[12]. In this paper, we deal with such link prediction issue: we analyze the dynamics of Interest Based Mobile Communities and we build our prediction model for users future movement by exploiting the abstraction level of users correlation patterns and their IMoComm.

The following of the paper is structured as follows. Section 2 states the problem. Section 3 presents the preprocessing of data set used in our work, and general statistics. In section 4, we introduce the prediction model based on community related features. In section 5 we discuss the results of experiments made on the available individual trajectories. Finally, conclusions are given in Section 6.

2 Problem statement

In daily life, people participate in various communities (colleagues, family, friends, ect). Their mobility is driven by their interest and need to practice different activities with other people depending on the type of the community they share (colleagues, friends, food, shopping, tourism, sport, ect). Hence, we study the human mobility behaviors from the perspective of network science, in particular the goal of this paper is to study how to use the knowledge gained from the IMoComm membership of each person and how it can be used to predict the community evolution (future links). Firstly, we perform an unsupervised task to extract the link pattern between people that distinguishes meaningful Interest Based Mobile Community structures and expresses the individual mobility behavior while sharing a common interest regularly or from time to time. We study then the link prediction problem using the resulting learning graphs, and we formulate our problem as follows: we start from a weighted graph $G(V, E)$ where V and E are sets of nodes and links resulting from the

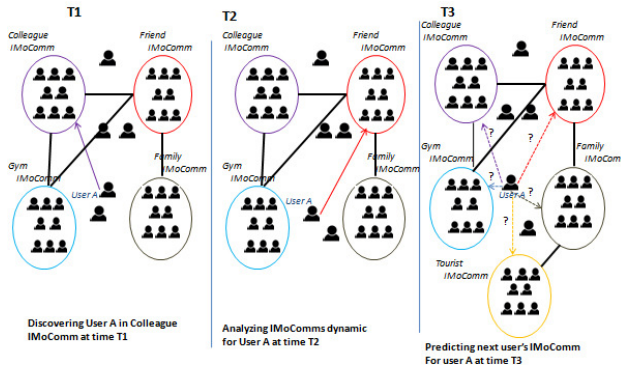


Fig. 1: An example to explain the link prediction problem in Interest Based Mobile Communities

IMoComm, respectively. Let the subgraph $G[t, t']$ denotes a snapshot of the social network between two times t and t' , such $t < t'$. We then predict the likelihood of a future connections between nodes and links in the network $G[t_1, t'_1]$. In other words, the link prediction aims to infer which new mobile community a user is likely to be at in the near future. So, predicting prospective links or deleted links in IMoComm graph for a future period is fundamental. Thus, we develop an approach to link prediction based on the analysis of community related features in the human mobility context.

A graphical representation is given in figure 1, in which solid links indicate that a user was already member of an IMoComm during the period $[t_0, t'_0]$, and dashed lines are used to indicate links that might appear during the interval $[t_1, t'_1]$ when users will move toward different communities.

3 Data set preprocessing

In our work, we use a very large dataset collected in GeoLife project and released by Microsoft Research Asia [13] [1]. The GeoLife dataset contains 2153 trajectories taken with different GPS loggers and GPS phones in different sampling rates and contains latitude, longitude and height of every sample. It contains 182 users and span a time period of five years from April 2007 till July 2012.

Human trajectories systems make use of location extraction techniques from geospatial data to identify locations that have meaning and importance to the users. Here, we have implemented stay points extraction method [6] in order to extract meaningful stay of individual who has spent a considerable time on a geographic region, for more details see [3]. The algorithm results in 23060 stay points for all users whose position is tracked in 2009. Figure 2 shows the total number of stay points per week in 2009 and illustrates the number of users having an accurately users' tracked position during 2009, we remark the lack of some users' traces, although we have generated missing data using the algorithm proposed in [2]. Besides, we limit our analysis to GPS data

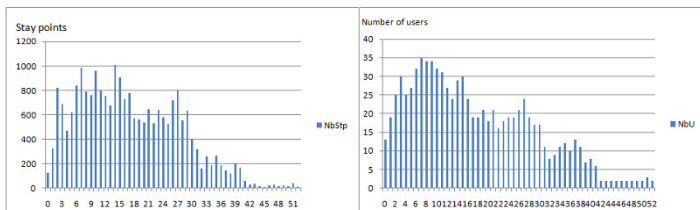


Fig. 2: a) Number of stay points per week in 2009, b) Number of users per week in 2009

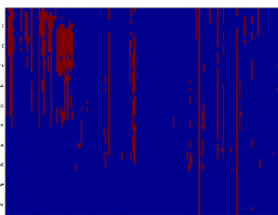


Fig. 3: Users movements during 52 weeks

collected in the region around Beijing. However, the rate at which users provide a new location point is not constant and not all users are present on all days (see Figure 3). It is therefore reasonable to select the active periods for our experiments.

4 Location Prediction based on mobile communities

The goal of our model is to predict the next IMoComm with which the user is going to interact at a certain day of the week, exploiting the learned visiting behavior of the user, his/her daily activities, and his/her relation between some users that share a common interest.

4.1 User’s communities pattern extraction

In order to achieve our ultimate goal, we start by discovering a learning graph that will be used to predict future potential links. Firstly, we apply DBScan algorithm [4] on stay points, the clustering parameters have been defined empirically ($minPoints = 3$, $\epsilon = 0.02 dd$), and we add semantics of location which imply the activities being carried out in each cluster in order to understand the relationship between the geographic location and the users activities. Basically, the individuals activities history consists of a sequence of couples of cluster-activity, thus

$$ExpU = (r_1, a_1) \rightarrow (r_1, a_2) \rightarrow \dots (r_l, a_2) \rightarrow \dots \rightarrow (r_l, a_q) \tag{1}$$

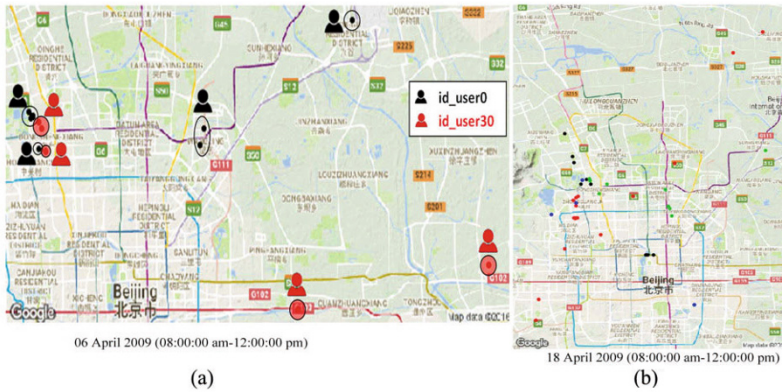


Fig. 4: Illustration of users similarities

where r_j is the j^{th} cluster covering a number of activities $a_i \in B$, $B = \{a_1, a_2, a_3, \dots, a_K\}$, that a users does in his/her stay locations. We have shown [3] that interesting locations for people can be grouped in several categories (regular activities, food activities, exceptional activities, Shopping activities, and Tourism activities). This suggests that people share common POIs (points of interest where a user has been at), but even more they share common interests. We thus mine the frequent behaviour of users with similar interests [3] and compute the similarity of two users in terms of similar activities at similar places using Ecludien distance between $UExp_i$ and $UExp_j$. From figure 4 a) we see that user of $id = 0$ and user $id = 30$ had revolved around similar POIs in physical places, during 06/04/2009 for a timestamp from 08 : 00 : 00am to 12 : 00 : 00pm, which result in social ties weighted with similarity value equal to 0.49 and these users have formed one IMoComm, especially in the specific timeline (timeline of work, or timeline of different daily activities). Figure 4 b) shows correlation between users who visits the same tourist places in weekend. Furthermore, we find an appropriate set of time intervals for the set of users since individuals are most likely to belong at group in a given time step, and apply community detection algorithm [8] on each snapshot and matching communities applying the algorithm described in [9], therefore we find a fundamental community structure and extracted features about development of human mobility and their IMoComm over time, we will discuss this finding in section 5.1.

In this phase, we have created the learning graphs $G_i = (V_i, E_i), \forall i \in 1..n$, describing the connection between mobile users, with V_i the set of nodes representing users and E_i the set of edges, which distinguishes meaningful Interest Based Mobile Community structure $C_{M_j}(V_{M_j}, E_{M_j}), \forall j \in 1..f$.

4.2 Community related features

The community information provided by the detected IMoComm provide powerful features for predicting individual's daily behaviors which are largely dependent on

his/her preferences, his/her activities and his/her social relations. As a matter of fact, for link prediction in location based networks, we should consider community features, which have interesting impact on the user's mobility:

- two users are regularly linked (strongly related) within a community are more likely to visit, in near future, the same IMoComm than two users who have no contact and/or don't share similar interest in similar place (weakly connected).
- Besides in communities that they attend regularly (such as communities of colleagues, friends, family, etc.), users exhibit a large similarities with the members of other communities they belong to, such communities are clearly more likely to affect more the behaviour than a community visited occasionally.
- Some groups have weakly connected links, thus their members are very varying over time (different community members in each snapshot).
- The more the same individuals form a community several times in specific period, the more regular and social the community is considered to be.

Our model accommodates these aspects as discussed in the following sections.

4.3 Prediction Model

Many methods for link prediction based on structural similarity between nodes have been proposed since similar nodes are likely to have neighbors in common and they are more likely to have the same relations in the near future[7]. Therefore, in our work, we have used a topological measure for weighted graph to calculate the likelihood score of any pair of nodes u and v

$$score_{(u,v)}^{Weighted} = \frac{\sum_{w \in \Gamma(u) \cap \Gamma(v)} \sqrt{A_{uw} * A_{vw}}}{\sum_{w \in \Gamma(u)} A_{uw} + \sum_{w \in \Gamma(v)} A_{vw}} \quad (2)$$

Where $\Gamma(u)$ is the set of direct neighbors of node u in $G[t_0, t_0]$, $\Gamma(u) \cap \Gamma(v)$ is the set of common neighbors of two nodes u and v in $G[t_0, t_0]$.

The prediction function P that indicates likelihood of nodes (u, v) being in E_{new} can be used for ranking all possible edges according to their probability.

$$P_{(u,v)} = \frac{\sum_{i=1}^{nbrday} score_{(u,v)}^{Weighted}}{nbrday} \quad (3)$$

The number of days $nbrday$ is set according to the selected training intervals.

Our prediction method is based on the assumption that human mobility is affected not only by person's travel experience but also by his/her movement towards Interest-Based Mobile Communities. Initially, we divided the extracted pattern in two parts, $G_{learning}$ and G_{test} , respectively, and select a learning period. For a sequence of snapshot $\langle G_1[t_0, t_0], G_2[t_1, t_1], \dots, G_l[t_l, t_l] \rangle$, in a given learning period, we compute the *probability list* for each missing links or links to occur in future. The algorithm

computes the likelihood of nodes for each temporal graph and then generates the whole graph applying an aggregation steps. The use of a graph aggregation produces the overall structure of the underlying graph during the learning period and captures semantic knowledge not only about individual nodes and their connections but also about groups of related nodes. Thus, we can recognize the times a node has appeared in a community over time in order to make a decision about its community type. For example, given two nodes u and v that belong to the same regular community reg_1 , their link (u, v) has a strong chance to appear in next time. If these two nodes belong to different regular communities reg_1, reg_2 the link might be formed in future time. Finally, whereas if these two nodes belong to the same or two different occasional communities oc_1 or oc_1, oc_2 respectively, they do not have a strong priority, and the link between them is most likely won't occur in next period. Using community attributes helps in predicting the IMoComm that will may be visited during his/her next move. Finally, the algorithm takes the global probability list and sort it in decreasing order of the likelihood $P(u, v)$ and of the community types features. So, the k links in the top are most likely to exist.

Algorithm 12 Link-Prediction Algorithm

Data: $\langle G_1[t_0, t'_0], G_2[t_1, t'_1], G_3[t_2, t'_2], \dots, G_l[t_l, t'_l] \rangle, id_{user} \in U$

Output: $PredictList, L$

- 1: Select the learning graphs: $G_1[t_0, t'_0], G_2[t_1, t'_1], \dots, G_{l-1}[t_m, t'_m]$
 - 2: **for all** $G_i[t_i, t'_i]$ **do**
 - 3: $A \leftarrow A_{G_i}$
 - 4: compute $score_{u,v}^{Weighted}$
 - 5: read (H) ▷ History of user's communities
 - 6: **end for**
 - 7: $aggregate(G[t_0, t'_0], G[t_1, t'_1], \dots, G_{l-1}[t_m, t'_m])$
 - 8: compute $P_{(u,v)agg}$
 - 9: $CommunityTypes(u) \leftarrow generate(assign(CommunityTypes, id_u))$
 - 10: $CommunityTypes(v) \leftarrow generate(assign(CommunityTypes, id_v))$
 - 11: compute $E_{wrong}, E_{positive}$ in $G_{l-1}[t_l, t'_l]$
 - 12: $PredictList \leftarrow Insert(P(u, v), id_u, id_v, CommunityTypes(u), CommunityTypes(v))$
 - 13: Sort $PredictList$ in descending order of the likelihood $P(u, v)$ and of the community types
 - 14: $L \leftarrow$ Get top k links in $PredictList$
 - 15: Validation using $G_l[t_l, t'_l]$ for test
-

5 Experiments

5.1 Communities and mobility

In order to study and predict the dynamics of individuals and investigate their communities evolution in human mobility domain, it is essential at first to identify the evolution characteristics of this complex network in particular the occurrence of new links and duration of interaction of their entities. For example, citation networks have a small number of evolution steps that is a snapshot per year, while the biological networks have more and specific details of evolution. To this end, we have made several empirical tests to distinguish the dynamic features and the social aspects related to the evolution of IMoComm:

1. The time step for each snapshot have to be taken for different timing that are closely related to the nature of the individuals interactions and their daily activities (according to the time-line of works, food, meeting friends, ect).
2. Relevant communities are created from the aggregation of all the links that appear and reappear at least twice during successive week days, for different timing, for all the studied period. This link type corresponds to regular human interactions, such as interactions between colleagues in work. This aspect is present in the blue and the red communities illustrated in Figure '5.
3. If two individuals perform the same activity in the same place only once, we consider that they are weakly connected and their link is not presented in the aggregation graph. However, we can take into account such links if they belong to public group and if they will help to characterize social aspect of human mobility. Moreover, this link type belongs sometimes to occasional IMoComm that exhibit a dense local structure around public places (such as tourist and cultural places). As evidence of such property, we have extracted a dense subgraphs during some weekend days; we have $Q = 0.393$ during 12/04/2009, $Q = 0.534$ during 19/04/2009, and $Q = 0.33$ in 26/04/2009.
4. The detection of IMoComm from the aggregate graph increases modularity and allows to identify a set of relevant communities.
5. The detected disjoint subgraphs in steps of daily time permit to discover overlapped communities on the aggregate graph. This is due to the fact that, in daily life, individuals can belong to multiple IMoComm but their number remains limited (daily communities). Figure 5 illustrates the disjoint groups for which we have selected few users who have continuous data collected for at least three consecutive days.

5.2 Prediction results

To predict a link, we select a training period and first extract the topological and community features for the temporal graphs, and then build the prediction model. Hence, given the selected temporal graphs $G_1[06/04, 10/04]$, $G_2[13/04, 18/04/]$,

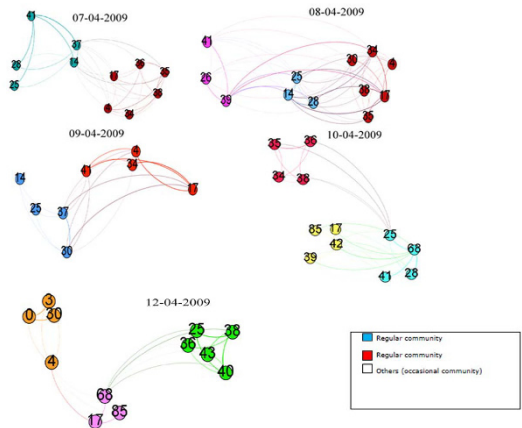


Fig. 5: Dynamic of IMoComm over four days (from 07/04/2009 to 10/04/2009) and a day of a week, the time period is from 08:00:00 am to 12:00:00 pm

$G_3[20/04, 25/04]$, we partition them in training and test sets. The choice of intervals has been made in an empirical way. We denote the training interval to be 11 days: $[06/04, 10/04]$ and $[13/04, 18/04/]$. We take $G_2[13/04, 18/04/]$ for labeling and we check that each pair (u, v) either represents a positive example (link exists) or a negative example (link does not exist). Thus the test graph $G_3[20/04, 25/04]$ is used to validate if a link exist or not(see Table 1).

From our dataset, we combines three datasets with different characteristics: *Dataset1* considers only working days, *dataset2* includes a weekend day (Saturday), while *dataset3* includes four weekend days.

Table 1: Training and test periods for link prediction for three datasets

Datasets	Phase	Period	Edges	Nodes	Comm	Temporal sequence of graphs
Dataset1	Training phase	From: 01/04/2009 to: 10/04/2009	346	182	03	$G_1[01/04, 03/04]$, $G_2[06/04, 10/04]$, $G_3[13/04, 17/04]$
	Testing phase	From: 13/04/2009 to: 17/04/2009	212	182	03	
Dataset2	Training phase	From: 06/04/2009 to: 18/04/2009	420	182	04	$G_1[06/04, 10/04]$, $G_2[13/04, 18/04]$, $G_3[20/04, 25/04]$
	Testing phase	From: 20/04/2009 to: 25/04/2009	246	182	04	
Dataset3	Training phase	From: 10/04/2009 to: 25/04/2009	598	182	04	$G_1[10/04, 19/04]$, $G_2[20/04, 25/04]$, $G_3[26/04, 30/04]$
	Testing phase	From: 26/04/2009 to: 30/04/2009	112	182	03	

At the community level, our approach allow us to recognize the expected user’s communities at the next step and to understand how a user plans his/her next move from his/her IMoComm’s perspective. As we can see from Figure 6, where we have used *dataset1*, mobility history of user’s communities extracted with our approach indicates that, despite the diversity of their travel history, humans follow, in most of

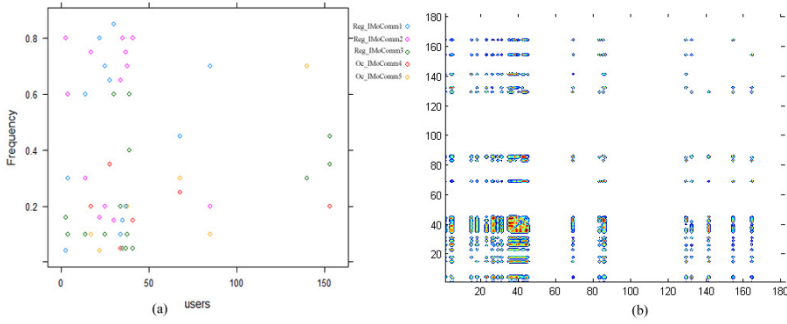


Fig. 6: a) Distribution of mobility history of user’s communities extracted when analyzing the dynamic of IMoComm b) Prediction of users’ future link and thier expecting IMoComm from 13/04/2009 to 17/04/2009)

case, simple reproducible pattern and have small number of communities. Thus the prediction process, which recognizes the most frequent communities of individuals travels, can predict movement of users and allows to characterize the common mobility behavior within his/her groups in the near future.(see Figure 6 b)). Figure 7 a) shows the type of predicted links between users using the IMoComm based approach. For instance, the probability that user $id_{user} = 38$ will join his/her regular community ($Reg_{IMoComm2}$) is equal to 0.446, and he/she may also move to $Reg_{IMoComm3}$ at the next step with probability 0.369, while probability to move to $OC_{IMoComm5}$ is 0.163 and therefore it is not selected in the predicted list. This confirms that an individual move usually to some of his/her regular IMoComm, while it is unlikely that he/she will go to some occasional communities.

Thus, the proposed approach is very useful for predicting users mobile behavior with regard to his/her next IMoComm. However, the algorithm needs more users attributes to be able to recognize the formation of new groups in near future. We mention also the problem of matching communities in dynamic complex networks which is an NP-Hard problem that we don’t study it here.

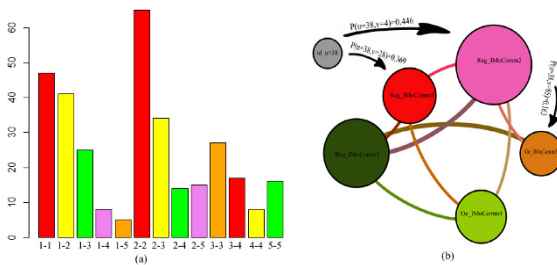


Fig. 7: a) Community type for users ’future links b) Illustration of next IMoComms for user 38

To evaluate the proposed approach we use precision, recall, and F-measure evaluation metric as performance measure for link prediction which is defined as follows:

$$Precision = \frac{E_r}{E_{predict}} \tag{4}$$

$$Recall = \frac{E_r}{E_{predict-positive}} \tag{5}$$

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

Where E_r , $E_{predict}$, and $E_{predict-positive}$ represent the corrected predicted links, the total predicted links, and the positive predicted links, respectively.

Our evaluation, for the three datasets shows that the average recall is 0.0.66, the average precision is 0.53, and the average F-measure is 0.59 (see figure 8). In *dataset1* we have an accurate prediction expressed by the recall equal to 0.66; the high precision (0.53) indicates significant prediction, thus we see that individuals exhibit regularity of belonging to their regular communities, and this community type is qualified as stable groups which appear and reappear in precise timing. The experiment with *dataset2* reveals still notable prediction performance with a small decrease compared to dataset1. This is due to the formation of irregular communities during the weekend days, which generates improbable links in our model for the next users movement. Due to the presence of large number of weekends, in *dataset3* the movements of each single user do not appear as continuous. Therefore, in this case, it is more efficient to use other users attributes extracted from their complex networks (social media, transportation networks, etc) in order to improve the prediction of the next users movement and his/her IMoComm either regular or occasional one.

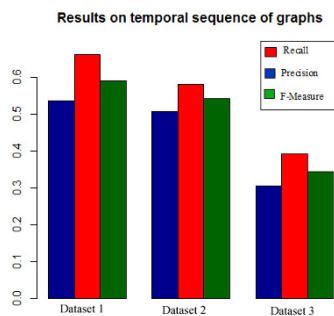


Fig. 8: Performance of the prediction method for three social graphs during one month

6 Conclusion

In this work, we have consider the problem of designing a link prediction model for location-based services. We have analyzed the dynamic of individuals at community level over different timing and thus have defined communities prediction features. We take advantage of these user's patterns and we therefore have investigated user's Interest Based Mobile Communities to reduce prediction space and then predict user mobility. In order to further improve the accuracy of the proposed prediction method, we are planning as future work to improve our model based on more consistency community related features and using several users attributes extracted from his/her multiple complex networks.

References

- [1] GeoLife GPS trajectories (2010). URL <https://www.microsoft.com/en-us/download/details.aspx?id=52367>
- [2] Baraldi, A.N., Enders, C.K.: An introduction to modern missing data analyses. *Journal of School Psychology* **48**(1), 5–37 (2010)
- [3] Drif, A., Boukerram, A., Slimani, Y., Giordano, S.: Discovering interest based mobile communities. Tech. rep., NetLab, ISIN-DTI, SUPSI, Switzerland (2016)
- [4] Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Kdd*, vol. 96, pp. 226–231 (1996)
- [5] Garg, K., Papandrea, M., Giordano, S.: Users'?? locations visiting prediction algorithm based on community mobility and users'?? interest profiling. Tech. rep., NetLab, ISIN-DTI, SUPSI, Switzerland (2016)
- [6] Li, Q., Zheng, Y., Xie, X., Chen, Y., Liu, W., Ma, W.Y.: Mining user similarity based on location history. In: *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, p. 34. ACM (2008)
- [7] Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. *Journal of the American society for information science and technology* **58**(7), 1019–1031 (2007)
- [8] Lusseau, D., Newman, M.E.: Identifying the role that animals play in their social networks. *Proceedings of the Royal Society of London B: Biological Sciences* **271**(Suppl 6), S477–S481 (2004)
- [9] Meyer, D., Leisch, F., Hornik, K.: Adaptive information systems and modeling in economics and management science. *Benchmarking Support Vector Machines* (2002)
- [10] Pang, J., Zhang, Y.: Location prediction: communities speak louder than friends. In: *Proceedings of the 2015 ACM on Conference on Online Social Networks*, pp. 161–171. ACM (2015)
- [11] Wang, D., Pedreschi, D., Song, C., Giannotti, F., Barabasi, A.L.: Human mobility, social ties, and link prediction. In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1100–1108. ACM (2011)
- [12] Wang, P., Xu, B., Wu, Y., Zhou, X.: Link prediction in social networks: the state-of-the-art. *Science China Information Sciences* **58**(1), 1–38 (2015)
- [13] Zheng, Y., Zhang, L., Xie, X., Ma, W.Y.: Mining interesting locations and travel sequences from gps trajectories. In: *Proceedings of the 18th international conference on World wide web*, pp. 791–800. ACM (2009)

Part IV
Dynamics on Networks

Why Amicus Curiae Cosigners Come and Go: A Dynamic Model of Interest Group Networks

Dino P. Christenson and Janet M. Box-Steffensmeier

Abstract Interest groups use coalition strategies to exert influence, yet, like other political actors, they also withdraw from partnerships in the pursuit of other policy goals. We explore how interest group coalition strategies have changed over time and which factors determine whether interest groups relationships form and dissolve. Utilizing dynamic networks of a panel of interest groups derived from cosigner status to United States Supreme Court amicus curiae briefs, we illuminate the evolution of the social networks of frequent signers from the 1970s to the present day. A separable temporal exponential random graph model (STERGM) shows that the number of partners is important for formation but not dissolution, while industrial homophily helps both to make and maintain connections. In addition, statistical trends suggest that while networks change, a few players have acted continuously as coordination hubs for the bulk of the decades. However, a number of other key players in particular decades would be missed without a dynamic perspective.

1 Interest Group Coalition Strategies

It is common knowledge that interest groups use coalition strategies. That is, interest groups, like other political actors, create ties with each other and demonstrate their working relationships in pursuit of mutually beneficial policy goals. Yet, many questions remain about such coalitions, particularly with regards to their historical development and over time dynamics. Most importantly, perhaps, little is known about the maintenance of relationships among interest groups. Though there is a modicum of work on the factors that draw interest groups together, few, if any, explore the factors of dissolution. In this work we seek to provide a more comprehensive

Dino P. Christenson (e-mail: dinopc@bu.edu)
Boston University, 232 Bay State Road, Boston, MA 02215

Janet M. Box-Steffensmeier (e-mail: steffensmeier.2@osu.edu)
The Ohio State University, 2140 Derby Hall, 154 N. Oval Mall, Columbus, OH 43210

account of interest group coalition dynamics by investigating both their development *and demise*.

Classic works in the interest group literature have sought to understand why interest group coalitions form. The dominant perspective is that coalitions serve as an economical and efficient means to form a more powerful bloc [e.g., 1, 2, 19, 20, 29, 32]. Coalitions signal broad support to policy makers on an issue [13, 21, 23, 25]. Thus, some factors thought to drive coalition formation are perceived strength of the opposition, previous experience in a coalition, and whether the group is critical to the success of the coalition [19].

Social network theory also suggests that alliances form out of the pursuit for access to resources and information [14]. That is, coalitions function as pipelines through which information and knowledge flow. The incentive for interest groups to form networks appears to be similar to that of firms: to diffuse information more quickly and benefit from the efficiency of cooperation [14, 15, 31, 32]. In addition, groups can benefit from the kinds of control offered in coalitions, such as sanctions, reputation, and trust. From this perspective, interest group network formation is largely a purposive act [18] for shared survival [26, 27]. Via the pooling of their resources and the creation of networks groups exhibit their shared policy preferences and divide the costs. In sum, the literature suggests that the motivations for coalitions among interest groups are plentiful, as are the rewards. The positive effects of networks on group performance has been demonstrated in terms of growth [28], speed of innovation [16], organizational learning [17], and reputation [30].

However, there is also good reason to expect interest groups to prefer to work alone—or, at the very least, work only sparingly in coalitions. Interest groups must maintain some autonomy from the other groups in their coalition, or risk losing their identification and competitive advantage. Thus interest groups have to consider coalitions in light of the need for differentiation. Groups would like to be seen as different enough to attract and maintain a constituency despite wanting to cooperate when they believe it will be helpful to attain valued resources. Such is at the foundation of economic theories of organizational behavior [33]. Interest groups require a niche to maintain their existence.

Collective action is thus a delicate balance. Interest groups benefit from sharing resources and signaling broad support to the targets of their pressure. However, interest groups must also demonstrate unique features that make them particularly appealing and allow them to claim credit for their accomplishments to their constituencies. Ultimately, this dance between cooperation and differentiation suggests that interest groups should not always pursue coalition strategies, but, instead, only do so when they find it necessary to accomplish their goals. As such, we expect interest group coalitions not to be permanent, with partnerships dissolving and perhaps even reappearing over time. In what follows, we engage a dynamic perspective to explore interest group networks and evaluate factors that may lead interest groups to dissolve or maintain their coalitions.

2 Hypotheses of Formation & Dissolution

The dynamic approach to interest group networks focuses on the potential for new ties to form and old ones to fall apart. This should hold true in the case of those with prior ties as well as those without them, so-called isolates, or “lone wolves” [5]. Just as all partnerships are not permanent, solitary behavior in the past does not necessarily lead to it in the future. We expect new ties to develop between organizations both with a history of working in coalitions and with a history of going it alone.

While ties may come and go it is unlikely that prior coalition behavior will be completely unrelated to future behavior. That is, we might expect those interest groups that have used coalitions in particular ways to try to do so again. In particular, organizations known to play the role of a hub or “team leader” [5] early on may be more likely to do so again. Likewise, organizations that work in large/small coalitions at time t are more likely to be those that do so again at time $t + 1$. As such, and despite some expected changes in networks over time, there is good reason to expect persistent roles for many of the organizations.

Interest groups may form coalitions based on a host of resource factors and common interests, which implies that these coalitions are not totally inclusive. Interest groups are selective about who they work with, and thus we posit that there will be limits to the number of partners for any group. As opposed to a pure contagion effect that we might see in other networks (e.g., campaign donors), we expect that for each additional partner the probability of adding another partner will decrease. We similarly test to see whether more partners leads to greater persistence of the network.

Finally, we would like to understand whether organizational attributes have similar effects on network formation and dissolution. In particular, some work distinguishes types of interest groups, arguing that different types of interest groups are more or less likely to join coalitions [9, 10]. This suggests that one should account for the type of interest group, such as whether it is a trade association, citizen group, or union. While this distinction is not statistically significant in all cases [23], there is recent evidence that working in the same industry draws groups together [5, 6]. There is less reason to believe that industry area should maintain those relationships. While working in the same industry might lead to introductions and first attempts at coalition building, maintaining the relationship might depend on other factors, like a previously good encounter. In sum, we expect the effects of industry area to be of greater importance in formation than dissolution.

3 Comparing Static & Dynamic Networks

The underlying networks of interest groups are difficult to perceive. It is widely acknowledged that they exist, but interest groups are unlikely to be perfectly forthcoming about their coalition partners and contacts in organizations during interviews or in surveys, as their livelihood may depend to some extent on restricted access to their partners and confidentiality among them [3, 12, 24].

In order to study interest group networks, we utilize the Amicus Curiae Network database [4]. This data set includes all the interest groups that have signed onto an amicus curiae brief from 1930 to the present, which amounts to more than 15,000 unique organizations over nearly 9 decades. We use cosigning on a brief, a “purposive and coordinated” political action, to join organizations in a network [5]. In Supreme Court cases, various parties with related interests submit briefs to the Court in favor of the petitioner, respondent, or in some cases, neither. Frequently, these signers are comprised of interest groups [11]. Groups frequently coordinate on the content of a brief and cosign with one another.

The analyses in this paper makes use of a small subset of the amicus network data. In order to look at changes in organizations’ partnerships over time we rely on a panel of repeat signers. The 167 organizations in our analyses signed onto at least one brief in every decade since the 1970s. Per usual, we use cosigning on these briefs to create ties between interest group nodes, but here we do so for each decade, thereby arriving at a five wave panel of interest group networks.

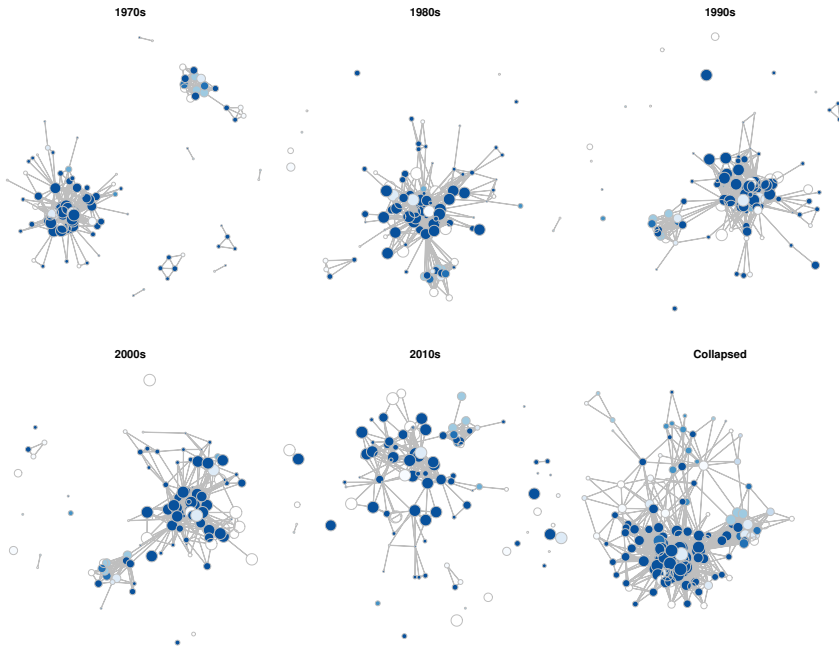
We begin by comparing the decade networks with a static network collapsed over all five decades. Graph structure in one or more of the decades that does not resemble that of the static network would suggest that it may be fruitful to explore the factors of network formation and persistence with dynamic models. Figure 1 plots both the decade networks as well as the single collapsed network plot. In terms of the latter, each node in the plot refers to a unique organization and an edge is drawn between organizations that cosigned a brief together at any time in the last five decades. Node size is proportional to the number of edges and color refers to the industry, as classified by the major divisions of the Standard Industrial Classification (SIC) code [5, 7]. Collapsing over the decades presents a dense network of primarily service organizations with only 8 isolates.

In terms of the decade networks, for each decade we have included the same 167 organizations but only drawn ties between groups that signed together in that decade.¹ The node size refers to the degree in the first decade, the 1970s, while the color again refers to the SIC code. While many of the large nodes are consistently central in the graphs, the fact that we see a number of large nodes in the periphery of the post-1970s graphs suggests that central groups in the 70s do not always remain so in subsequent decades. That is, the highly connected groups in one decade may not be the same as those in other ones. In short, comparing the collapsed network plot with the decade networks suggests that there may be good reason to look at network dynamics instead of a static network.

To give us a clearer idea of what is happening to the edges in the dynamic network, the left graph in Figure 2 plots the panel slices against the timeline of edges, one horizontal line for each edge. When the horizontal line corresponding to a tie between two organizations in one period crosses the vertical line associated with the panel period, the edge would be included in that network. Thus each panel period (e.g., 1 to 2) corresponds to a social network created in that period. Lines that carry over to the next panel period (e.g., 2 to 3) means that that tie remained through the next period

¹ We also provide a video of the organizations changing ties over each decade at <http://dinopc.tumblr.com/#121184498222>.

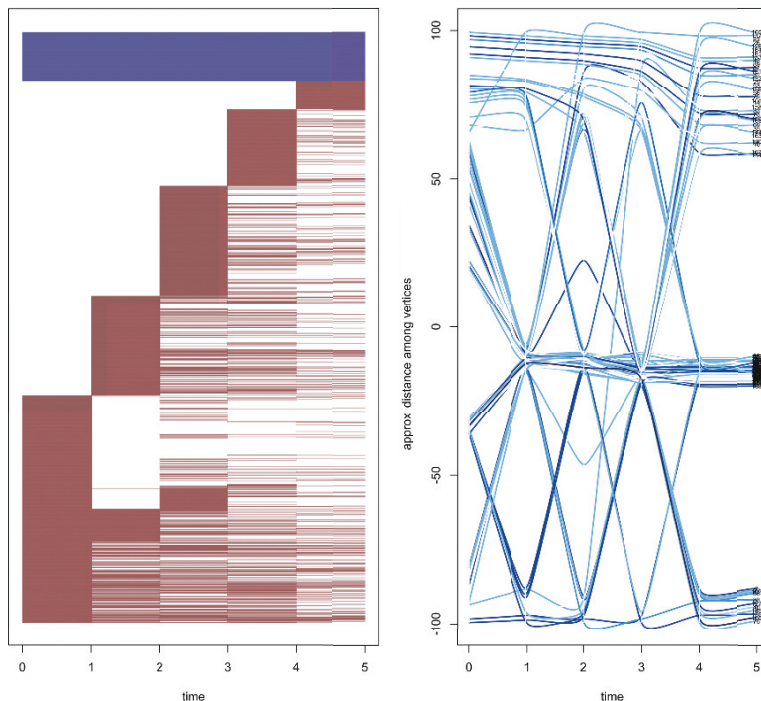
Fig. 1: Interest Group Networks by Decade and Collapsed



in time. In the Figure we see all of the ties in the starting period, 0 to 1. Looking from 1 to 2 we see that only about half of those ties remain in the next with a host of new ties appearing in that period as illustrated by the new solid block of ties a step above the initial block of ties. The solid set of lighter colored lines at the top show that several ties, only about a fifth of the organizations, remain from the first to the last period. The increasing lightness of the graph as you move from left to right illustrates that many new ties form across time and many dissolve as well, which suggests that there is good reason to explore the amicus curiae network as dynamic.

Graphing the timing of edges is helpful in revealing the dynamic density of events. However, it tells us little about the panel to panel changes in network structure and connectivity. For that we rely on the graph on the right side of Figure 2. It illustrates the overall shifts in the network by collapsing the momentary structure to a single vertical dimension and plotting across time. Here, for each panel we calculate the geodesic distance and plot the vertices' distances with each vertex's position in each panel linked by a spline [8]. Thus this figure provides a horizontal trajectory of a vertex as a line, with tightly connected vertices situated close to one another. Curves moving up or down illustrate the group to group movement while flat lines illustrate stability. The color again refers to the SIC code. The Figure shows that while some partnerships remain, there is substantial changes in the network structure

Fig. 2: Timing of Edges & Proximity Timeline



in every period of the panel. Moreover, neither stability nor change are restricted to organizations in the same industry.

Given the dynamics in the structure of these networks, we should expect that the roles of some of the groups in these networks are ephemeral. That is, a group that is particularly well connected or essential to the quick transmission of information in one period may not be so in the next. Looking solely at the collapsed network may hide various temporarily powerful players. We gain insight into the coalition behavior of these interest groups by looking at the best connected, highest degree, as well as those on the shortest path between groups, highest betweenness for both the collapsed and decade networks.

As shown in Table 1, the most connected organizations in the collapsed network are the American Civil Liberties Union (ACLU), Legal Momentum, American Jewish Committee, National Council of Jewish Women and the National Women's Law Center. With the exception of the first organization, it is important to recognize that the distribution of degree changes gradually. That is, in this network there is a wide range of different numbers of edges across the nodes, with just about everything between 0, for the eight isolates, to 68, for the second most connected group, Legal Momentum. The ACLU also appears among those organizations on the shortest

path to others, along with the National Association of Criminal Defense Lawyers (NACDL) and the National Association of Manufacturers.

Table 1: Top 5 Highest Scores on Centrality Measures

Degree		Betweenness	
<u>Collapsed</u>			
Am. Civil Liberties Union	82	Equal Employment Advisory Council	761
Am. Jewish Committee	67	Am. Civil Liberties Union	1895
Natl. Council of Jewish Women	66	Natl. Assoc. of Criminal Defense Lawyers	967
Natl. Womens Law Center	64	Natl. Assoc. of Manufacturers	1084
Legal Momentum	68	Natl. School Boards Assoc.	645
<u>1970s</u>			
Mex. Am. Legal Defense & Educ. Fund	40	Am. Civil Liberties Union	496
Natl. Council of Jewish Women	40	Mex. Am. Legal Defense & Educ. Fund	247
Natl. Council of the Churches of Christ US	40	Natl. Council of the Churches of Christ US	162
Natl. Organization for Women Foundation	41	Natl. Education Assoc.	192
Legal Momentum	41	Legal Momentum	185
<u>1980s</u>			
Am. Civil Liberties Union	54	Am. Civil Liberties Union	2636
Mex. Am. Legal Defense & Educ. Fund	32	Natl. Assoc. of Criminal Defense Lawyers	617
Am. Jewish Committee	35	Natl. Wildlife Federation	699
Natl. Education Assoc.	34	Anti-Defamation League	719
Legal Momentum	32	Planned Parenthood Federation	896
<u>1990s</u>			
Am. Assoc. of University Women	45	Internat. Assoc. of Chiefs of Police	584
Am. Civil Liberties Union	50	Am. Civil Liberties Union	2921
Am. Jewish Committee	45	Natl. Assoc. of Broadcasters	1317
Natl. Council of Jewish Women	48	Natl. Assoc. of Manufacturers	952
Natl. Womens Law Center	45	Anti-Defamation League	590
<u>2000s</u>			
Am. Civil Liberties Union	48	Am. Civil Liberties Union	2495
Mex. Am. Legal Defense & Educ. Fund	35	Natl. Assoc. of Criminal Defense Lawyers	1077
Natl. Assoc. of Social Workers	39	Natl. Trust for Historic Preservation	756
Natl. Council of Jewish Women	38	Legal Momentum	898
Natl. Education Assoc.	35	Pacific Legal Foundation	849
<u>2010s</u>			
Am. Assoc. of Retired Persons	21	Chamber of Commerce of USA	1356
Natl. Organization for Women Foundation	19	Am. Medical Assoc.	1189
Legal Momentum	22	Am. Assoc. for Justice	2540
Union for Reform Judaism	20	Am. Assoc. of Retired Persons	1416
Am. Assoc. for Justice	24	Natl. Assoc. of Criminal Defense Lawyers	1085

Looking at the centrality measures in the decade-by-decade networks in Table 1 we arrive at a somewhat familiar list of organizations. The ACLU, the National Council of Jewish Women, Legal Momentum, and National Education Association (NEA) make frequent appearances as highly connected in the decade networks. The ACLU has a similarly high presence as an informational bridge between other organizations, appearing in the top betweenness in a few of the decades, as does the NACDL. However, the static network also undervalues a number of important players in specific periods. For instance, the decade networks show that the National Organization for Women (NOW) were particularly connected in the 1970s, and the NEA in the 1980s and 2000s and the American Association of Retired Persons (AARP) in the 2010s. Likewise, the Mexican American Legal Defense and Educational Fund had high information control in the 1970s, the National Wildlife Federation (NWF) in the 1980s, the National Association of Broadcasters in the 1990s, as well as the Chamber of Commerce and AARP in the 2010s.

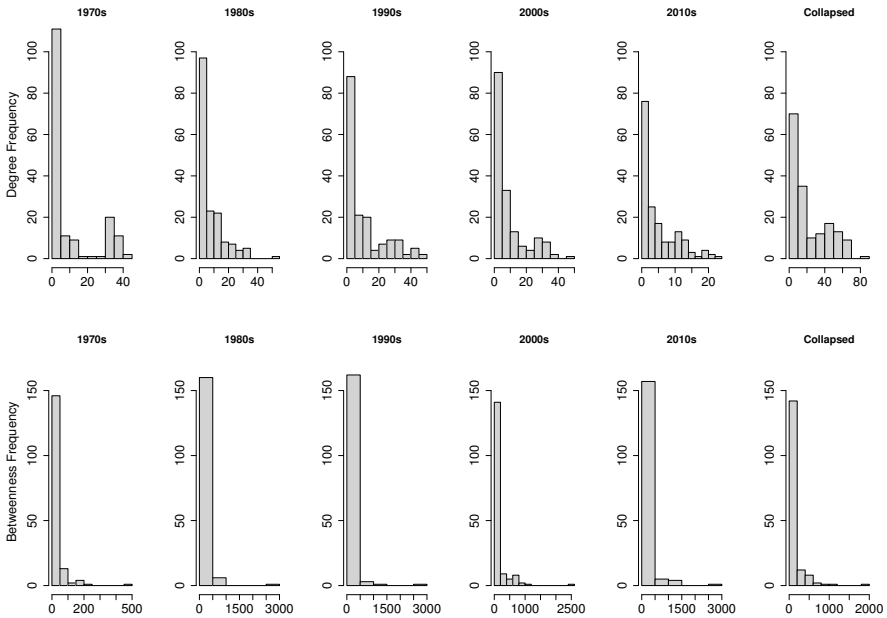
Figure 3 shows the distribution of the degree and betweenness measures from both the decades and collapsed networks. The collapsed network is presented in the last column of the Figure. Degree centrality shows primarily bimodal shaped distributions with a larger amount of organizations huddled in the lower portion of the graph. That is, there is an abundance of organizations with few connections and a small portion with many in most decades and in the collapsed network. The bimodal plots in the 1970s, 2000s and 2010s appear most similar to that of the collapsed network. The less pronounced right tail in the 1980s conveys a smaller than usual number of highly connected organizations. The distributions on betweenness shows less variance with the bulk of organizations having low information control, since most exist within cliques and few are uniquely positioned on shortest path connections to other organizations. The plots show the 1970s as having an unusually low number of high betweenness organizations.

The centrality results above show that the static and dynamic networks share a number of characteristics, but not all. The network, edge timing and proximity timeline graphs show that new ties develop over time and old ties are not permanent. Both sets of results suggest the value of a dynamic approach. However, we still have little understanding how these relationships come about and what leads to their demise or perseverance. To those ends, we turn below to a stochastic model to explore the effects of both structural and node level factors on network formation and dissolution.

4 Stochastic Model Results

Separable temporal exponential random graph models (STERGM) extend the familiar ERGM for dynamic networks in discrete time [22]. The methodological innovation allows us to model the formation of new ties between interest groups as well as their perseverance. Recall that the ERGM provides a single model of static network formation. STERGMs, however, combine two ERGMs to model both the relational formation and dissolution. The formation and dissolution ERGMs work similarly to

Fig. 3: Histograms of Network Centrality by Decade and Collapsed



the standard ERGM, except here there exists a time index to the tie values as well as a conditional statement that differs for the formation and dissolution equations. The formation equation is conditional on a tie not existing between interest groups in the previous period. The dissolution equation is conditional on the tie existing. The STERGM then combines the respective equations. Estimation is performed via conditional maximum likelihood (CML).

Table 2 presents the results of the STERGM. Given our hypotheses, we similarly specify the formation and dissolution parameters in the STERGM. Pertaining to our hypotheses on the number of shared partners we specify both edges and degree terms. The edges term adds a single statistic for the number of edges in the network. Degree adds a statistic for each of the nodes with the relevant number of degrees. Thus degree 0 takes into account the isolates. In order to test the hypotheses of organizational attribute homophily, we also add a statistic to the model for each set of joined nodes that share an industrial area. Again, we do so for both the formation and dissolution stages to test whether organization attributes previously shown to influence network development also affect network persistence.

We consider the formation and dissolution models together for each parameter to emphasize the similarities and differences in the factors of formation and dissolution. The negative edges parameter can be interpreted similarly to an intercept in a logit model. It suggests that the conditional log-odds of two organizations forming a tie

Table 2: STERGM of Interest Group Networks

	Formation	Dissolution
Edges	-3.16*** (0.04)	-0.38*** (0.06)
Degree 0	7.89*** (0.30)	0.51* (0.21)
Degree 1	4.93*** (0.28)	0.28 (0.20)
Degree 2	4.40*** (0.20)	0.21 (0.19)
SIC Homophily	0.44*** (0.05)	0.25*** (0.07)
Num. vertices	668	668
AIC	12040.19	4121.18
BIC	12084.53	4151.27

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

would be -3.16 , provided the tie does not add any statistics for homophily or the specified degrees. The negative probability of tie formation, holding constant at zero the other parameters, is noticeably smaller for dissolution.

The decreasing in magnitude yet consistently positive coefficients on the degree terms means that there is an underlying tendency for relational formation to occur, which continues to at least two partners, though the effect is reduced with each pre-existing tie that the two organizations are involved in. That is, there is a strong incentive to be in a relationship with one and two other organizations. However, dissolution appears to be largely independent. Existing relationships have a similar underlying dissolution probability at every point in time.

Perhaps most interestingly from a social science perspective, the attribute homophily shows consistently positive effects in the formation and dissolution models. Though the effect is much greater for the former, meaning that working in the same industry area brings interest groups together, working in the same industry area also makes a tie more likely to persist.

5 Conclusion

This work has the potential to provide a number of contributions to the literature on interest group behavior. Foremost, the interest group coalitions of the most frequent players in the modern era are not perfectly stable. While many of the most central players are fairly consistent throughout time, some key players are limited to particular decades. Moreover, the shape, size and overall structure of networks ranges substantially. New relationships develop and old ones dissolve.

We also provide evidence that the development and dissolution of interest group coalitions are driven by different factors. Interest groups feel the need to share resources and demonstrate large support via coalitional work, which brings interest groups to work with more than one partner. However, the number of partners matters little for maintaining the network in subsequent periods. We also find that industry homophily plays a stronger role in the formation of networks than it does in maintaining them. Still, the evidence here suggests that shared interests both bring groups together and keep them that way.

References

- [1] Berry, J.M.: *Lobbying for the People: The Political Behavior of Public Interest Groups*. Princeton University Press (1977)
- [2] Berry, J.M., Wilcox, C.: *The Interest Group Society*. Longman (1989)
- [3] Box-Steffensmeier Janet M, D.P.C., Leavitt, C.: *Oxford Handbook of Political Networks*, chap. Judicial Networks. Oxford University Press (2016)
- [4] Box-Steffensmeier, J.M., Christenson, D.P.: The amicus curiae networks database **Version 1** (2012)
- [5] Box-Steffensmeier, J.M., Christenson, D.P.: The evolution and formation of amicus curiae networks. *Social Networks* **36**, 82–96 (2014)
- [6] Box-Steffensmeier, J.M., Christenson, D.P.: Comparing membership interest group networks across space and time, size, issue and industry. *Network Science* **3**(1), 78–97 (2015)
- [7] Box-Steffensmeier, J.M., Christenson, D.P., Hitt, M.P.: Quality over quantity: Amici influence and judicial decision making. *American Political Science Review* **107**(3), 1–15 (2013)
- [8] Butts, C.T., Leslie-Cook, A., Krivitsky, P.N., Bender-deMoll, S.: *networkDynamic: Dynamic Extensions for Network Objects* (2013). R package version 0.5
- [9] Caldeira, G.A., Wright, J.R.: Amici curiae before the supreme court: Who participates, when, and how much? *The Journal of Politics* **52**(3), 782–806 (1990)
- [10] Clark, P.B., Wilson, J.Q.: Incentive systems: A theory of organizations. *Administrative Science Quarterly* **6**(2), 129–166 (1961)
- [11] Collins, P.M.: *Friends of the Supreme Court: Interest Groups and Judicial Decision Making*. Oxford University Press (2008)
- [12] Cummings, J.: *New disclosure reports lack clarity* (2008)
- [13] Esterling, K.M.: *The Political Economy of Expertise: Information and Efficiency in American National Politics*. Ann Arbor: University of Michigan Press (2004)
- [14] Gilsing, V., Nooteboom, B., Vanhaverbeke, W., Duysters, G., van den Oord, A.: Network embeddedness and the exploration of novel technologies: Technological distance, betweenness centrality and density. *Research Policy* **37**(10), 1717–1731 (2008)
- [15] Gilsing, V.A.: *The Dynamics of Innovation and Interfirm Networks: Exploration, Exploitation and Co-evolution*. Edward Elgar (2005)
- [16] Hagedoorn, J.: Understanding the rationale of strategic technology partnering: Interorganization modes of cooperation and sectoral differences. *Strategic Management Journal* **14**(5), 371–385 (1993)
- [17] Hamel, G.: Competition for competence and inter-partner learning within international strategic alliances. *Strategic Management Journal* **12**, 83–103 (1991). Special Issue: Global Strategy
- [18] Hathaway, W., Meyer, D.S.: The Lessons of the Nuclear Freeze, chap. Competition and Cooperation in Movement Coalitions: Lobbying for Peace in the 1980s. Rienner (1997)

- [19] Hojnacki, M.: Organized interests' advocacy behavior in alliances. *Political Research Quarterly* **51**(2), 473–459 (1998)
- [20] Hula, K.: Rounding up the usual suspects: Forging interest group coalitions. In: A.J. Cigler, B.A. Loomis (eds.) *Interest Group Politics*, 4 edn. CQ Press (1995)
- [21] Kingdon, J.W.: *Congressmen's Voting Decisions*, 2 edn. Harper and Row, New York (1981)
- [22] Krivitsky, P.N., Handcock, M.S.: A separable model for dynamic networks. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **76**(1), 29–46 (2014)
- [23] Mahoney, C.: The power of institutions: State and interest-group activity in the european union politics. *European Union Politics* **5**(4), 441–466 (2004)
- [24] Mayer, L.R.: Under the radar. Published online at OpenSecrets: Capital Eye Blog (2007)
- [25] Mayhew, D.R.: *Congress: The Electoral Connection*. Yale University Press, New Haven (1974)
- [26] McCarthy, J.: *Social Movements in an Organizational Society*, chap. Pro-Life and Pro-Choice Mobilization: Infrastructural Deficits and New Technologies. Transaction Books, New Brunswick, NJ (1987)
- [27] McCarthy, J., Zald, M.: Resource mobilization and social movements: A partial theory. *American Journal of Sociology* **82**, 121–1241 (1977)
- [28] Powell, W.W., Koput, K.W., Smith-Doerr, L.: Interorganizational collaboration and the locus of innovation: Networks of learning in biotechnology. *Administrative Science Quarterly* **41**(1), 116–145 (1996)
- [29] Schlozman, K.L., Tierney, J.T.: *Organized Interests and American Democracy*. Harper and Row (1986)
- [30] Stuart, T.E.: Network positions and propensities to collaborate: An investigation of strategic alliance formation in a high-technology industry. *Administrative Science Quarterly* **43**(3), 668–698 (1998)
- [31] Teece, D.J.: Profiting from technological innovation: Implications for integration, collaboration, licensing and public policy. *Research Policy* **15**(6), 286–305 (1986)
- [32] Whitford, A.B.: The structures of interest coalitions: Evidence from environmental litigation. *Business and Politics* **5**(1), 45–64 (2003)
- [33] Wilson, J.Q.: *Political Organizations*. Princeton University, N.J. (1973)

Contradictory information flow in networks with trust and distrust

Giuseppe Primiero, Michele Bottone, Franco Raimondi and Jacopo Tagliabue

Abstract We offer a proof system and a NetLogo simulation for trust and distrust in networks where contradictory information is shared by ranked lazy and sceptic agents. Trust and its negative are defined as properties of edges: the former is required when a message is passed bottom-up in the hierarchy or received by a sceptic agent; the latter is attributed to channels that require contradiction resolution, or whose terminal is a lazy agent. These procedures are associated with epistemic costs, respectively for confirmation and refutation. We describe the logic, illustrate the algorithms implemented in the model and then focus on experimental results concerning the analysis of epistemic costs, the role of the agents' epistemic attitude on distrust distribution and the influence of (dis)trust in reaching consensus.

1 Introduction

Trust of information transmissions facilitates reliability and enforces security in networks. This applies in particular to hierarchical structures, e.g. in access control models [2, 3, 14], and where reputation is at work, e.g. in social networks [4, 9, 18]. Trust and distrust on communication channels are also affected by the agents' epistemic attitude, their ability and willingness to check information and their readiness to reject it. Negative trust has recently become a topic of interest in computational contexts [11, 13]. In particular, understanding conditions of (dis)trust propagation and the costs related to topological and epistemic factors is crucial for dynamic (social) network analysis and access control models [1, 6, 8, 21], with applications in mathematics, computer science, economics and biology. Negative accounts of trust

Giuseppe Primiero (e-mail: G.Primiero@mdx.ac.uk) · Michele Bottone (e-mail: M.Bottone@mdx.ac.uk) · Franco Raimondi (e-mail: F.Raimondi@mdx.ac.uk)
Department of Computer Science, Middlesex University London

Jacopo Tagliabue (e-mail: tagliabue.jacopo@gmail.com)
AXON VIBE, New York

are essential especially for networks that allow contradictory information diffusion but require coherent agents.

In this paper we offer a logic and a NetLogo simulation for networks with contradictory information and where agents identify their channels as trustful or distrustful. Our agents are qualified as sceptic or lazy and are given an initial ranking depending on the topological features of the network. The network is seeded initially with two items of contradictory information $(p, \neg p)$. Each node is labelled by either piece of data, with a resolution procedure when both are received by the same node. At each step, the node assigns a trust or a distrust property to the relevant edge. In our experimental analysis we consider in particular:

1. the epistemic costs of trust and distrust according to different network topologies;
2. the distrust distribution in view of the epistemic attitude of the seeding agents;
3. the role of distrust in reaching consensus.

The paper is organized as follows. In Section 2 we overview related work. In Section 3 we introduce the calculus $(Un)SecureND^{sim}$ which includes rules for trust and distrust. In Section 4 we provide the principles underlying the graph construction and algorithm design at the basis of the simulation. In Section 5 we describe our experimental results. Finally, Section 6 presents general observations on our analysis and shortly illustrates future work.

2 Related Work

In reporting on previous work, we focus in particular on three different aspects: controversial users vs. controversial trust values; binary and continuous trust values; local vs. global trust methods.

In [12] controversial users are those generating a disagreement on their trustworthiness, either as the minimum between trust and distrust evaluations by other users, or as the difference in the number of trust and distrust judgements. [20] considers instead controversial trust values between two nodes, determined either as the trust weight of their edge, or as a fixed negative value when no path exists, or as a continuous value $t \in [0, 1]$ when there is no direct edge. Similarly, in our logic trust is a function on formulas obtained by verification, mimicked in the network model by a property of edges when a node is labelled.

Differently from the above, our model uses discrete values but it combines the comparative ranking of agents with both their epistemic attitudes and a majority selection in the case of conflicting information. [12] also uses a binary classification for users, so do several models for belief diffusion in social networks, with binary opinions for agents, considering neighbours' influence [5, 9] or majority ([18]). Continuous models, on the other hand, might depend on the weight of other agents' opinion [10] or admit influence only below a certain distance [7].

Trust defined by global methods is a value attached to a user and appropriate for a reputation evaluation at network level; in local methods, trust is inferred instead as a value between source and sink nodes, i.e. it is an edge feature. As it appears

clearly from the above, our approach uses a local trust method in the case of non-conflicting information, resorting to a computation of trust path lengths to determine which elements need to be distrusted in the case of conflicting information. This combination of features recalls the two controversial cases discussed in [20]: the *ToTrustOrNotToTrust* case resembles our binary choice, but moderated by continuous trust values, while we rely on ranking and epistemic attitudes; the *Asymmetric Controversy* case resorts to path lengths with preference for shortest paths, while we base our result on the number of distrustful edges present in each path.

To the best of our knowledge, no other work in the current literature combines a rule-based semantics with ranked agents with epistemic attitude, using local trust values with path length analysis for the resolution of contradictory information.

3 (Un) SecureND^{sim}

The natural deduction calculus SecureND [16] is a logic designed for secure operations on resources issued by subjects with different privileges; it guarantees trusted content checked for consistency at every transmission. (Un) SecureND [15] is an extension with negation to model two forms of negative trust. In [17], the calculus SecureND^{sim} is adapted to model contradictory information propagation under trust in a network of ranked agents and is simulated in NetLogo [19]. In this contribution we present (Un) SecureND^{sim}, extending the previous system to deal with a distrust function. We refer to a set of agents as V and an individual agent as v_i . Agents behave differently in the context of information transmission:

- *sceptic agents* and *agents reading from below in the hierarchy* require verification when receiving a message, and as a result they trust the related channel;
- *lazy agents* and *all agents in the presence of contradictions* have a rejection attitude, with the result of distrusting the related channel.

Verification and rejection are computationally costly processes for the agents and these costs are tracked in our model.

Definition 3.1. The syntax of (Un)SecureND^{sim} is defined by the following alphabet:

$$\begin{aligned}
 V &:= \{\text{lazy}(v_i), \text{sceptic}(v_i)\} \\
 BF^V &:= p^{v_i} \mid \neg p^{v_i} \\
 \text{mode} &:= \text{Read}(BF^V) \mid \text{Verify}(BF^V) \mid \text{Write}(BF^V) \mid \text{Trust}(BF^V) \mid \text{DisTrust}(BF^V) \\
 RES^V &:= BF^V \mid \text{mode} \mid \neg RES^V \\
 \Gamma^V &:= \{\phi_1^{v_i}, \dots, \phi_n^{v_i}\};
 \end{aligned}$$

V is the set containing lazy and sceptic agents; BF^V are literals, i.e. atoms and their negations; in the following, when needing a metavariable for either, we will use ϕ^V ; mode is for access functions over atoms; RES^V includes both contents and access modes, with negation. In line with standard notation for natural deduction, we use Γ^V to express a context of expressions (typed by one agent in V , and feasible to extension to another agent's context) in which a given formula is derivable: such a context

$$\begin{array}{c}
\frac{\Gamma^{v_i} \vdash wf}{\Gamma^{v_i}; \Gamma^{v_j} \vdash \phi^{v_j}} \text{Atom} \quad \frac{\Gamma^{v_i} \vdash mode(\neg\phi^{v_j})}{\Gamma^{v_i} \vdash \neg mode(\phi^{v_j})} \neg\text{-distribution} \\
\\
\frac{\Gamma^{v_j} \vdash wf \quad \Gamma^{v_i} \vdash \phi^{v_i}}{\Gamma^{v_i}; \Gamma^{v_j} \vdash Read(\phi^{v_i})} \text{read_down} \\
\\
\frac{\Gamma^{v_i}; \Gamma^{v_j} \vdash Read(\phi^{v_i}) \quad \Gamma^{v_j}; \phi^{v_i} \vdash wf}{\Gamma^{v_j} \vdash \phi^{v_j}} \text{read_elim} \\
\\
\frac{\frac{\Gamma^{v_i} \vdash Read(\phi^{v_j})}{\Gamma^{v_i} \vdash Verify(\phi^{v_j})} \text{verify_high} \quad \Gamma^{v_i} \vdash \phi^{v_i} \quad v_j \in \text{sceptic_node}}{\Gamma^{v_j} \vdash Verify(\phi^{v_i})} \text{verify_sceptic} \\
\\
\frac{\Gamma^{v_i} \vdash Verify(\phi^{v_j}) \quad \Gamma^{v_i}; \phi^{v_j} \vdash wf}{\Gamma^{v_i} \vdash Trust(\phi^{v_j})} \text{trust} \\
\\
\frac{\Gamma^{v_i} \vdash Read(\phi^{v_j}) \quad \Gamma^{v_i} \vdash Trust(\phi^{v_j})}{\Gamma^{v_i} \vdash Write(\phi^{v_j})} \text{write_trust} \\
\\
\frac{\Gamma^{v_i} \vdash \phi^{v_i} \quad \Gamma^{v_i} \vdash Read(\neg\phi^j)}{\Gamma^{v_i} \vdash \neg Verify(\neg\phi^{v_j})} \text{unverified_contra} \\
\\
\frac{\Gamma^{v_i} \vdash Read(\phi^{v_j}) \quad v_i \in \text{lazy_node}}{\Gamma^{v_i} \vdash \neg Verify(\phi^{v_j})} \text{unverified_lazy} \\
\\
\frac{\Gamma^{v_i} \vdash \neg Verify(\phi^{v_j})}{\Gamma^{v_i} \vdash DisTrust(\phi^{v_j})} \text{distrust} \quad \frac{\Gamma^{v_i} \vdash DisTrust(\phi^{v_j})}{\Gamma^{v_i} \vdash Write(\neg\phi^{v_j})} \text{distrust_elim}
\end{array}$$

Fig. 1: The system (Un)SecureND^{sim}

matches the graph G of agents introduced in the next section; the derived formula matches a new labelled vertex added to the graph. Formulas of this language are of the general form $\Gamma^{v_i} \vdash RES(\phi^{v_j})$, saying that an agent v_i accesses under her profile a message ϕ originated by agent v_j . Access is here neutral for all the operations included in *mode*. An order relation \leq over $V \times V$ models the dominance relation between agents: $v_i \leq v_j$ means that agent v_i has equal or higher priority (e.g. in terms of security privileges) than agent v_j .

The rules system (Un)SecureND^{sim} is introduced in Figure 1 and it assumes that $v_i \leq v_j$ holds. This logic allows the following operations. Any content is accessible within a well-formed (*wf*) user profile (*Atom*). Accessing a negation of a content implies that the contrary cannot be accessed (\neg -distribution): this is a strong

negation rule, justifying the resolution procedure for contradictions. Any content can be read from agents downwards in the order relation (*read_down*) and it is accepted if it preserves the profile consistency (*read_elim*). Reading by an agent upwards in the dominance relation or by a sceptic agent is possible by invoking a verification procedure (*verify_high* and *verify_sceptic* respectively). Such verification checks consistency and then applies a trust function on the object of the message (*trust*). Reading and trusting guarantee rights to write formulae (*write_trust*). The remaining rules define the behaviour of distrust relations. Reading contradictory information or reading by a lazy agent induce a rejection procedure (*unverified_contra* and *unverified_lazy* respectively). This in turn means that a distrust operation is executed (*distrust*), and the opposite message to the one read can be written (*distrust_elim*).

4 Model Design and Implementation

The network is an undirected graph $G = (V, E)$, with a set $V = \{v_i, \dots, v_n\}$ of vertices (agents) and a set $E = \{e_{(i,j)}, \dots, e_{(n,m)}\}$ of edges (information transmission channels). A labelled node $v(p)$ denotes an agent knowing p ; $v(\neg p)$ denotes an agent knowing $\neg p$; $v()$ is used for a vertex with no label and denotes an agent who does not hold any knowledge yet. An edge between two nodes is fully denoted by $e(v_i(), v_j())$ with the appropriate labels: $e(v_i(p), v_j())$ expresses a channel from i to j such that the former can transmit p over. A non-standard notation with three nodes $e(v_i(p), v_j(), v_k(\neg p))$ is used to abbreviate the fact that the following edges exist: $e(v_i(p), v_j())$ and $e(v_j(), v_k(\neg p))$ and it requires a resolution procedure. When need for reference to multiple vertexes arises, we shall use the notation $v_{i,\dots,n}$. The order relation among nodes is total or partial in view of the network topology. In a total network, each vertex has an edge connecting it to any other vertex and all have equal ranking; the underlying dominance relation is then a total order. In the linear network, each vertex has an edge to the next vertex higher in the ranking; by transitivity, also this order is total. In the random network, by introducing a new node at least one edge with another vertex is established; the ranking is here assigned by the seeding node and never overwritten, the order is partial. The scale-free network model uses the Barabasi-Albert method: it is initialised by $m = 3$ nodes and each node v_j without neighbours is connected to up to $n < m$ existing vertices with a probability $p_{v_j} = \frac{k_{v_j}}{\sum_{v_i} k_{v_i}}$, where k_{v_j} is the number of neighbours of agent v_j and the sum is made over all pre-existing nodes v_i . Newly added nodes tend to prefer nodes that already have a high number of links. The ranking in this case is given as $\frac{1}{|edges|}$. The maximum number of vertices in our graphs is set at 300.

The randomly seeded contradictory information ($p, \neg p$) flows in the network, according to the algorithm *Transmission* in Figure 2. If the receiving agent is sceptic or a non-contradictory message comes from below in the dominance relation, a successful transmission is preceded by a sub-routine *Verify*, described in Figure 3. The latter implies an epistemic cost, the new node is successfully labelled and the edge is qualified as trusted. If the receiving agent is lazy, a new subroutine *Distrust* is executed, by which the edge is qualified as distrusted and the related

```

1  PROCEDURE Transmission( $G$ ), with  $\phi \in BF^V$ 
2
3   $G := (V, E)$ 
4
5  FOR  $e(v_i(\phi), v_j()) \in G$ 
6    IF  $v_j() \in \text{sceptic}$  OR  $\text{ranking}(v_j()) < \text{ranking}(v_i(\phi))$ 
7      THEN  $\text{Verify}(e(v_i(\phi), v_j()))$  AND  $G' := G \cup (v_j(\phi))$ 
8    ELSEIF  $v_j() \in \text{lazy}$ 
9      THEN  $\text{Distrust}(e(v_i(\phi), v_j()))$  AND  $G' := G \cup (v_j(\neg\phi))$ 
10   ENDIFELSE
11 ENDFOR
12
13 FOR  $e(v_i(\phi), v_j(), v_k(\neg\phi)) \in G$ 
14    $\text{SolveConflict}(e(v_i(\phi), v_j(), v_k(\neg\phi)))$ 
15
16 RETURN  $\text{Trusted}(G)$ 
17 ENDPROCEDURE

```

Fig. 2: Algorithm for Simple Information Transmission

```

1  PROCEDURE Verify( $e(v_i(\phi), v_j())$ )
2
3    set COSTTRUST+1
4    set TRUSTLINK  $e(v_i(\phi), v_j(\phi))$ 
5    RETURN  $\text{Trusted}(G)$ 
6  ENDPROCEDURE

```

Fig. 3: Algorithm for Trust Costs Increase

```

1  PROCEDURE Distrust( $e(v_i(\phi), v_j())$ )
2
3    set COSTDISTRUST+1
4    set DISTRUSTLINK  $e(v_i(\phi), v_j(\neg\phi))$ 
5    RETURN  $\text{Trusted}(G)$ 
6  ENDPROCEDURE

```

Fig. 4: Algorithm for Distrust Costs Increase

epistemic costs are increased, Figure 4. A node receiving contradictory data $(p, \neg p)$ starts a resolution process SolveConflict , see Figure 5: it analyses the number of distrusted links appended to each neighbour with each contradictory piece of information and it selects the new label from the least distrusted one, proceeding by random choice (*) when an equal number of distrusted links is detected. It then executes the subroutine Distrust on the selected link.

5 Experimental results

The code for the simulation and all data from the experiments are available at <https://bitbucket.org/gprimiero/cn16>. The experiments have been executed on a machine with 7.7 GB of memory, 64bit Ubuntu 16.04 system, NetLogo

```

1  PROCEDURE SolveConflict( $e(v_i(\phi), v_j(), v_k(-\phi))$ )
2
3      let d1 #DISTRUSTLINK  $e(v_{i\dots n}(\phi), v_j())$ 
4      let d2 #DISTRUSTLINK  $e(v_{k\dots m}(-\phi), v_j())$ 
5
6      IF (length d1 > length d2)
7          THEN  $G' := G \cup (v_j(-\phi))$  AND Distrust( $e(v_i(\phi), v_j(-\phi))$ )
8      ENDIF
9
10     IF (length d1 < length d2)
11         THEN  $G' := G \cup (v_j(\phi))$  AND Distrust( $e(v_k(-\phi), v_j(\phi))$ )
12     ENDIF
13
14     IF (length d1 = length d2)
15         IF *
16             THEN  $G' := G \cup (v_j(-\phi))$  AND Distrust( $e(v_i(\phi), v_j(-\phi))$ )
17             ELSE  $G' := G \cup (v_j(\phi))$  AND Distrust( $e(v_k(-\phi), v_j(\phi))$ )
18         ENDIFELSE
19     ENDIF
20 ENDPROCEDURE

```

Fig. 5: Algorithm for Conflict Resolution

5.3. We have collected data from several synthetic networks of fixed dimensions between 10 and 300 nodes, with seeding of labels $(p, \neg p)$ randomly associated to two sceptic/lazy nodes. We consider first different network topologies and then focus on scale-free networks only, which better represent the topology of complex graphs as they occur for example in social networks. On the other hand, linear networks are more common in hierarchical structures that can be encountered in conditions of access control. In both cases, the role of trust and distrust operation is crucial to information propagation.

5.1 Costs of Trust/Distrust by Network Topology

In the first run of experiments we compare different network topologies of fixed size (300 nodes), each equipped with a fixed proportion of sceptic nodes (50%). We consider in particular the size of trusted and distrusted edges and the related costs for each topology.

As shown in Figure 6 and the associated Table, the average rate of links and costs is inversely proportional: the former increases from random, through linear, scale-free and total networks, while the latter decreases. Given the fixed number of sceptic agents across the various topologies, the decrease in costs should be mainly associated with the ranking of agents and their order, while the increase in trusted links is purely due to the number of links in the network. From these data it appears that random networks perform the worst, as the required costs are high but the obtained links are less than in scale-free or linear networks.

The different topologies show a similar pattern with respect to distrust values. As shown in Figure 7 and the associated Table of average values, random networks

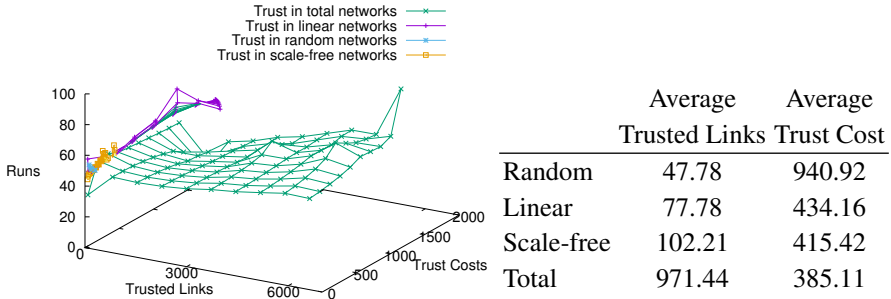


Fig. 6: Trust distribution and average costs

are the most expensive with respect to distrust, and have the lowest number of distrusted links; linear networks remain constrained in number of distrusted links, with costs decreasing; scale-free networks do not show a sensibly better behaviour, with comparable number of distrusted links and costs; finally, total networks perform the best, with the highest levels of links and relatively lower costs. As shown in the graph, it is remarkable the diverging behaviours of total and random networks: the former ones have almost stable distrust cost with increasing distrusted links, while the latter have stable links with increasing costs.

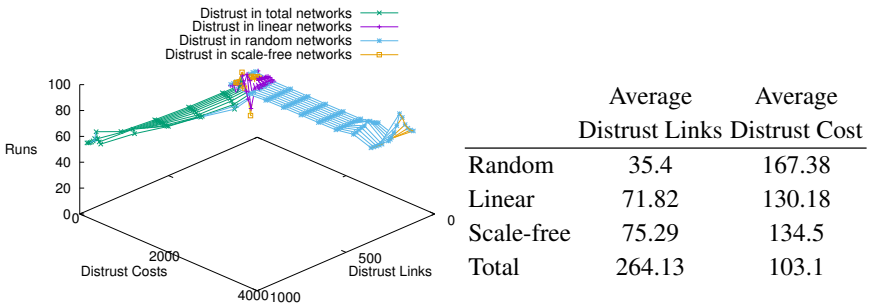


Fig. 7: Distrust distribution and average costs

The comparison between tables shows that the average number of trusted and distrusted links grows in parallel, while the related costs decrease in a similar vein across the different topologies. Trust propagates a lot more than distrust in these balanced networks, suggesting that the former is a more frequent and more relevant property in information transmission than the latter.

5.2 Distrust and epistemic attitude

In this and the following experiments, we focus on scale-free networks only and their distrust behaviour.¹ First, we consider distrust as a parameter of the proportion of lazy agents in a network of 300 nodes, with a random assignment of seeds to agents. As shown in Figure 8, there is a strict correlation between the proportion of sceptic and the distrust behaviour: the more lazy agents are present in the network, the higher its overall distrust value. While this is obvious in view of the algorithm design, it is interesting to remark that in the case of a fully sceptic network (where no lazy agents are allowed), the value of distrust is to be associated entirely with the presence of contradictory information, and hence it can be used as a parameter of contradiction diffusion. The associated Table offers average values over 100 runs. It illustrates that conflict resolution is responsible on average for roughly 10% of the network’s distrusted edges, with costs averaging at around $\frac{1}{7}$ of those of a highly lazy network (i.e. with 10% of sceptic agents).

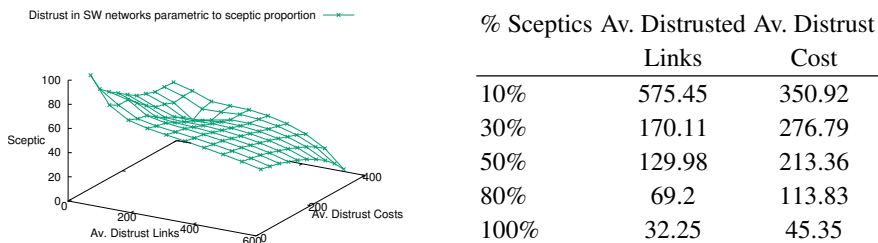


Fig. 8: Distrust behaviour and epistemic attitude.

We now extract the values for a balanced network (i.e. with 50% of sceptic agents) and compare them to the initial distribution of seeds qualified as lazy-sceptic agents. As Figure 9 shows, there is a strict correlation of the final distribution of distrust values with the initial condition of the network: the range of minimal values for both distrust costs and number of distrusted links is relatively stable, while their maximum values decreases when moving from a configuration that has two sceptic agents as initial nodes to one that has two lazy ones. The result on distrust across the network is less influenced by the role of agents *distributing* the information than by the role of agents *receiving* it.

5.3 Trust, Distrust and consensus

Our last experiment concerns the role of trust and distrust in reaching consensus. As shown in Figure 10, networks with trust *and* distrust present an inverse correlation

¹ For a more detailed analysis of further aspects of trust behaviour, see [17].

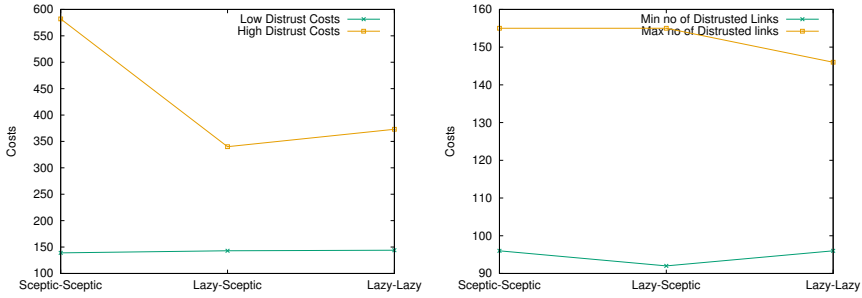


Fig. 9: Initial nodes' epistemic attitudes and distrust

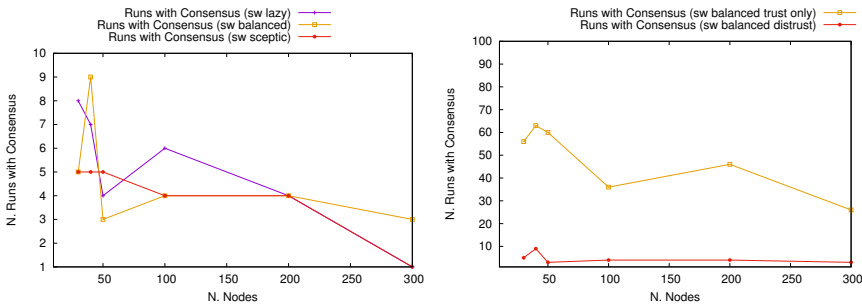


Fig. 10: Consensus in Scale-free Networks with distrust

between size and the number of transmissions that reach consensus: the smaller the network, more often full labelling with a unique formula is obtained (i.e. it is easier to reach consensus). Despite some differences in the reached peaks by lazy and balanced networks, the behaviour is overall similar in all configurations: balanced networks have the highest absolute number of such runs, while networks with higher proportion of sceptic agents have the lowest number of consensus reaching transmissions. Networks with distrust significantly differ from those with trust only for the total amount of consensus-reaching transmissions. We show this for balanced networks in the second graph of Figure 10, the same holding for lazy and sceptic networks: the presence of a distrust routine has a strong impact on the ability of the network to reach consensus in the presence of contradictory information, with no more than 9% of runs reaching a full labelling by either p or $\neg p$ (network of 40 nodes), while in the case of networks with trust only, this value reaches 63% (for networks of the same size).

6 Conclusions

We have presented a logic for the analysis of distrust propagation in a multi-agent system. We have offered related algorithms and an agent-based simulation of the

dynamics of such networks when transmitting contradictory information. Our initial experimental results, currently limited to synthetic networks and to be extended with real-world larger data sets, show that: distrust has a lower impact on information transmission in terms of costs than trust; it represents a strong obstacle to reaching consensus; and it qualifies up to a tenth of the size of the network in the presence of contradictory information. Further research will offer extensive comparison with other models, updates of epistemic attitudes and applications to swarm-like phenomena.

References

- [1] Carbone, M., Nielsen, M., Sassone, V.: A Formal Model for Trust in Dynamic Networks. In: A. Cerone, P. Lindsay (eds.) *Int. Conference on Software Engineering and Formal Methods, SEFM 2003.*, pp. 54–61. IEEE Computer Society (2003). URL <http://eprints.soton.ac.uk/262294/>. A preliminary version appears as Technical Report BRICS RS-03-4, Aarhus University
- [2] Chakraborty, S., Ray, I.: TrustBAC: Integrating Trust Relationships into the RBAC Model for Access Control in Open Systems. In: *Proceedings of the Eleventh ACM Symposium on Access Control Models and Technologies, SACMAT '06*, pp. 49–58. ACM, New York, NY, USA (2006). DOI 10.1145/1133058.1133067. URL <http://doi.acm.org/10.1145/1133058.1133067>
- [3] Chandran, S.M., Joshi, J.B.D.: LoT-RBAC: A Location and Time-based RBAC Model. In: *Proceedings of the 6th International Conference on Web Information Systems Engineering, WISE'05*, pp. 361–375. Springer-Verlag, Berlin, Heidelberg (2005). DOI 10.1007/11581062_27. URL http://dx.doi.org/10.1007/11581062_27
- [4] Grandi, U., Lorini, E., Perrussel, L.: Propositional Opinion Diffusion. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS '15*, pp. 989–997. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2015). URL <http://dl.acm.org/citation.cfm?id=2772879.2773278>
- [5] Granovetter, M.: Threshold models of collective behavior. *American Journal of Sociology* **83**(6), 1420–1443 (1978)
- [6] Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of Trust and Distrust. In: *Proceedings of the 13th International Conference on World Wide Web, WWW '04*, pp. 403–412. ACM, New York, NY, USA (2004). DOI 10.1145/988672.988727. URL <http://doi.acm.org/10.1145/988672.988727>
- [7] Hegselmann, R., Krause, U.: Opinion dynamics and bounded confidence models, analysis, and simulations. *Journal of Artificial Societies and Social Simulation* **5**(3), 2002
- [8] Jøsang, A., Pope, S.: Semantic Constraints for Trust Transitivity. In: S. Hartmann, M. Stumptner (eds.) *APCCM, CRPIT*, vol. 43, pp. 59–68. Australian Computer Society (2005)
- [9] Kempe, D., Kleinberg, J.M., Tardos, E.: Influential nodes in a diffusion model for social networks. In: *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (2005)*
- [10] Lehrer, K., Wagner, C.: *Rational Consensus in Science and Society*. D. Reidel Publishing Company (1981)
- [11] Marsh, S., Dibben, M.: Trust, Untrust, Distrust and Mistrust – An Exploration of the Dark(er) Side. In: P. Herrmann, V. Issarny, S. Shiu (eds.) *Trust Management, Lecture Notes in Computer Science*, vol. 3477, pp. 17–33. Springer Berlin Heidelberg (2005). DOI 10.1007/11429760_2. URL http://dx.doi.org/10.1007/11429760_2

- [12] Massa, P., Avesani, P.: Controversial Users Demand Local Trust Metrics: An Experimental Study on Epinions.com Community. In: Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA, pp. 121–126 (2005). URL <http://www.aaai.org/Library/AAAI/2005/aaai05-020.php>
- [13] McKnight, D.H., Chervany, N.L.: Trust and Distrust Definitions: One Bite at a Time. In: R. Falcone, M.P. Singh, Y. Tan (eds.) Trust in Cyber-societies, Integrating the Human and Artificial Perspectives, *Lecture Notes in Computer Science*, vol. 2246, pp. 27–54. Springer (2000). DOI 10.1007/3-540-45547-7_3. URL http://dx.doi.org/10.1007/3-540-45547-7_3
- [14] Oleshchuk, V.A.: Trust-Aware RBAC. In: I.V. Kotenko, V.A. Skormin (eds.) MMM-ACNS, *Lecture Notes in Computer Science*, vol. 7531, pp. 97–107. Springer (2012). URL <http://dblp.uni-trier.de/db/conf/mmmacns/mmmacns2012.html#Oleshchuk12>
- [15] Primiero, G.: A Calculus for Distrust and Mistrust. In: S.M. Habib, J. Vassileva, S. Mauw, M. Mühlhäuser (eds.) Trust Management X - 10th IFIP WG 11.11 International Conference, IFITM 2016, Darmstadt, Germany, July 18-22, 2016, Proceedings, *IFIP Advances in Information and Communication Technology*, vol. 473, pp. 183–190. Springer (2016). DOI 10.1007/978-3-319-41354-9_15. URL http://dx.doi.org/10.1007/978-3-319-41354-9_15
- [16] Primiero, G., Raimondi, F.: A typed natural deduction calculus to reason about secure trust. In: A. Miri, U. Hengartner, N. Huang, A. Jøsang, J. García-Alfaro (eds.) 2014 Twelfth Annual International Conference on Privacy, Security and Trust, Toronto, ON, Canada, July 23-24, 2014, pp. 379–382. IEEE (2014). DOI 10.1109/PST.2014.6890963. URL <http://dx.doi.org/10.1109/PST.2014.6890963>
- [17] Primiero, G., Tagliabue, J.: Quantifying epistemic trust in networks with contradictory information. Tech. rep. (2016)
- [18] Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorithm to detect community structures in large-scale networks. *Physical Reviews E* **76**(3) (2007)
- [19] Wilensky, U.: NetLogo, Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL. <http://ccl.northwestern.edu/netlogo/> (1999)
- [20] Zicari, P., Interdonato, R., Perna, D., Tagarelli, A., Greco, S.: Controversy in Trust Networks. In: Procs. 9th Int. Conf. on Trust and Trustworthy Computing (TRUST), Vienna, Austria, August 29-30, 2016, pp. 82–100 (2016). DOI 10.1007/978-3-319-45572-3_5
- [21] Ziegler, C.N., Lausen, G.: Propagation Models for Trust and Distrust in Social Networks. *Information Systems Frontiers* **7**(4-5), 337–358 (2005). DOI 10.1007/s10796-005-4807-3. URL <http://dx.doi.org/10.1007/s10796-005-4807-3>

The Echo Chamber Effect in Twitter: does community polarization increase?

Siying Du and Steve Gregory

Abstract A recent article criticized social media platforms for failing to mobilize society into action long enough to address any major global issue. This is attributed to the simplistic design of current social media platforms, which encourage ideas to spread virally but do not support consensus formation which might lead to lasting social change. One reason for this could be the well known echo chamber phenomenon, whereby people tend to discuss issues only with other like-minded people. Social media has been blamed for encouraging the echo chamber effect and increasing polarization in society. For example, in Twitter, it is very common for users to be followed by others with similar views. Is this a reflection of real life or does Twitter actually increase polarization of views? This paper investigates this by comparing the Twitter *follows* network at two points in time and detecting communities in the network of reciprocated *follows* relationships. We find that new edges are (at least 3-4 times) more likely to be created inside existing communities than between communities, and existing edges are more likely to be removed if they are between communities. This leads to the conclusion that Twitter communities are indeed becoming more polarized as time passes.

1 Introduction

A recent article [3] highlighted the paradox that, although the use of social media has becoming increasingly widespread, it has not been able to mobilize society into action long enough to address any major global issue. The authors blame this on the simplistic design of current social media platforms, pointing out the absence of mechanisms for reflection, argumentation, and consensus formation. This is related to the well known echo chamber phenomenon, whereby people tend to discuss issues only with others with similar views. Social media has been blamed for encouraging the echo chamber effect and increasing polarization in society [6, 7].

Siying Du · Steve Gregory (e-mail: steve@cs.bris.ac.uk)
Department of Computer Science, University of Bristol, Bristol BS8 1UB, UK

It is common knowledge that social networks, in real life as well as online, feature assortative mixing: people (or users) tend to communicate with those who are similar to themselves in some respect. When represented as networks, groups of vertices representing similar people tend to be more densely connected by edges than one would expect by chance [9]. This is the basis of community structure in networks, which has been studied intensively during the last 15 years [8].

In the context of online social media platforms, such as Facebook and Twitter, it is well known that user networks feature community structure. Users usually follow or friend other similar users, forming groups that are densely connected but loosely connected to other groups. When similarity is based on interests or opinions, users tend to be more strongly connected to others with similar interests and isolated from those with different interests or opposing viewpoints. One early study [1] analysed the network structure of (US domestic) political blogs and found that conservative and liberal blogs formed separate communities with little overlap. Following the launch of Twitter, another seminal work [5] obtained tweets related to a US election and constructed a *retweet* network, in which each edge represents a retweet from one user to another. This network was also found to split into two separate, ideologically opposed, communities.

The above works showed that social media platforms facilitate the echo chamber effect, by allowing users to form communities. However, this does not necessarily mean that these platforms encourage the formation of separate communities; they might have existed already.

The aim of this paper is to investigate whether social media platforms increase polarization of users, using Twitter as an example. We do this by checking whether community structure in the Twitter *follows* network becomes stronger, in some sense, as time passes. We consider only the network topology, ignoring the attributes of users and the content of their communication (tweets). We do not attempt to detect the topic or viewpoint that characterizes each community, or even verify whether a coherent topic exists. This is for simplicity and to avoid making our results dependent on a specific method of topic detection.

A naive approach might be to perform community detection [8] on the network and compute the modularity [10] of the partition, and repeat the process at different times. However, this would be impractical because

1. The Twitter *follows* network is too large to obtain and analyse, especially because access to it is rate-limited.
2. The network vertices change over time as users come and go.
3. Different partitions could be found each time, as an artefact of the (nondeterministic) community detection algorithm.
4. Modularity (or some other common metric) depends on many factors and would not reveal small changes in the strength of community structure.

Our approach avoids these problems, as follows:

1. We collect small samples instead of the whole (reciprocated) *follows* network.
2. We sample the same set of users each time the experiment is repeated.
3. We detect communities only on the first run of the experiment.

4. We measure the strengthening of the community structure by counting how many new reciprocated *follows* edges are created inside communities and how many edges are removed between communities, and comparing these with a null model in which edges are added and deleted randomly.

In the next section, we explain how data is collected from the Twitter network. Section 3 presents the experimental results for new and deleted edges, comparing these with a randomly changed network. Section 4 presents our conclusions.

2 Data collection

The data collection was done in two phases: in June and August 2016. In each phase, three network samples were collected. This section describes the network samples and how they were collected.

2.1 First phase: snowball sampling

The basic strategy for the first phase of data collection is snowball sampling. This starts from a seed user (vertex s) and crawls to all of its followers (users who follow s) and followings (or followees: users who are followed by s). This process is repeated recursively for each of the users found until enough vertices are obtained. We crawl to a maximum distance d from the seed, collecting all vertices at distance $0, 1, \dots, d - 1$, but not necessarily all vertices at distance d , because of the huge number of them.

In order to reduce the time costs, we choose a seed which has a reasonably small number of followers and followings. For our experiments we collected network samples from three seeds: a beauty blogger, a comic writer and a computer graphic scholar. We refer to these networks as Beauty, Comic, and Graphics, respectively.

2.2 Omitting users and edges

Because of the rate limit of Twitters API, which allows 15 requests every 15 minutes, it is time-consuming to collect users who have a large number of followings or followers. For example, if a user has 4 million followers, which is quite common for a famous person, it would take 13 hours to collect all of the users followers. Because of the time cost and the limited time available, it was necessary to restrict the data collection.

One way to achieve this would be to omit users who have a large number of followers, and the other is to partially collect the followers and followings of a user. Both of these methods will introduce bias to the data collected. For the first method, we might miss a user who is famous and has an important role within a community (as well as all edges of this user). Although the first method is not perfect, the bias of the second method is much more severe. If we were to omit some edges between users, we are likely to miss some users who would form triangles with other users

and create communities. For example, x , y , and z all follow each other, forming a triangle as shown in Fig. 1(a). If we partially collected followers of x and omitted z , which is a follower of x , the triangle $\{x, y, z\}$ might not be noticed, as in Fig. 1(b). As a result, the community detection might not place them into the same community, resulting in a distorted structure. Moreover, in this case, a deeper search might be needed to find z : in order to find z , one has to find y first. Obviously, the peripheral vertices will never be complete because the data collection has to stop somewhere, but we make sure that the network sample contains all edges for vertices that do appear in the sample. I.e., if the network sample is $G = (V, E)$ and $u \in V$ and $v \in V$ and $\{u, v\}$ exists in the complete network, then $\{u, v\} \in E$. Therefore, we decided to omit all users who have more than 50,000 followers.

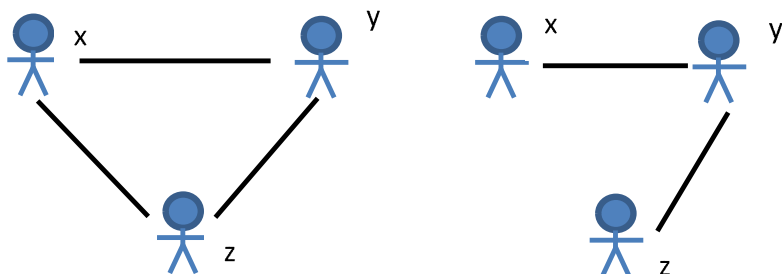


Fig. 1: (a) x , y , and z follow each other, forming a triangle. When collecting all followings and all followers of a user, this triangle can easily be found. (b) However, when collecting followings and followers partially, this triangle might be ignored.

2.3 Directionality

Considering the edge direction should be expected to contribute to a more accurate result [2]. However, in Twitter, any user u can follow any other user v , creating a directed edge (u, v) . Such a unidirectional edge is less valuable than a reciprocated pair of edges, u follows v and v follows u , which indicate a mutual relationship. We therefore focus on undirected networks, in which an edge $\{x, y\}$ means that x follows y and y follows x . When collecting followers of a specific user u , we omit those users that u does not follow; when collecting u 's followings, we omit users that do not follow u . To implement this, directed networks were collected and then converted to undirected networks with reciprocated edges after sampling.

2.4 Three datasets

In order to make our results more robust, we collected three different networks starting with three different seed vertices. Fig. 2 shows a visualization of the Graphics

network, while Table 1 shows some statistics about all three networks collected in the first phase, in June 2016. This describes the three directed networks and the three undirected networks which contain reciprocated edges only. Table 1 also shows the communities found by the Infomap algorithm [11] for each network. We use Infomap for all experiments in this paper because it is one of the best and most popular community detection algorithms.

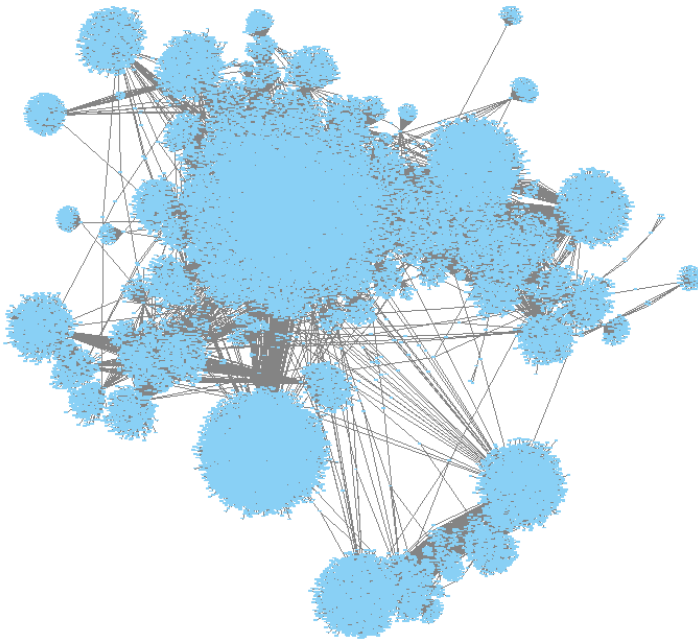


Fig. 2: Visualization of the Graphics network.

Table 1: Statistics of the three networks collected in June. The Directed columns indicate the vertices and edges before removing directionality. The Undirected columns describe the network of reciprocated edges.

Network	Directed		Undirected				
	vertices	edges	vertices	edges	density	communities	largest community
Beauty	6756319	10394337	249259	437852	1.4×10^{-5}	44	58275
Comic	2277503	3860175	101022	171990	3.4×10^{-5}	22	43681
Graphics	938960	1444554	47179	77909	7×10^{-5}	10	26563

2.5 Data collection for the second phase

There are two possible methods for the data collection of second phase (in August 2016). One is to crawl again from the same seed to collect a network by snowball sampling. The other is to directly collect all of the users that appeared in the first phase. Crawling from the beginning means doing a breadth-first search to a specific depth; this cannot ensure that all the users of first phase will be collected in the second phase. For instance, suppose that x , y , and z follow each other and form a triangle. When crawling from x with a depth of 1, this triangle will be found. However, if one of these edges is deleted before the second phase, a depth of 2 will be needed to find the triangle. As a result, crawling from the beginning with the same depth will omit some users that exist in the first phase, resulting in an incomplete network. Therefore, we chose to collect exactly the same users as in the first data collection phase, except those that no longer exist. Table 2 shows statistics about the same three network samples collected in August. (Note that, although we collect the same users as in the first phase, the number of vertices shown here is different because it includes all followers and followings.)

Table 2: Statistics of the three networks collected in August.

Network	Directed		Undirected	
	vertices	edges	vertices	edges
Beauty	6957428	10717644	248363	463797
Comic	2546530	4214677	103353	180604
Graphics	994529	1522011	47491	84028

3 Experiments

3.1 Edges of real network

Community detection was performed on the network samples from the second phase, but did not show any noticeable changes because two months is not enough time for communities to evolve. However, there are still a significant number of new edges and deleted edges. The next step is to investigate how often new edges appear inside communities, indicating that users start to follow others in the same community, and whether edges tend to be removed (by unfollowing) inside or between communities. We make two hypotheses:

1. New edges are more likely to appear inside communities than between communities.
2. Edges between communities are more likely to be removed than those inside them.

In the remainder of the paper, we refer to edges inside communities as *intracommunity* edges and edges between communities as *intercommunity* edges.

Fig. 3 shows the numbers of added and deleted edges of the three networks collected, counting only the edges between vertices that are present in both versions of the network. That is, we ignore vertices that existed only in the first snapshot, and their edges. For example, in the Beauty network, after two months, 5076 new edges appear: 3212 intracommunity edges and 1864 intercommunity edges. Similarly, in the other two networks, most of the new edges are intracommunity edges, which seems to support the first hypothesis stated above. For the deleted edges, for all three networks, the number of intracommunity deleted edges exceeds the number of intercommunity deleted edges, which seems to disprove our second hypothesis. However, intracommunity edges are far more numerous than intercommunity edges, so whenever an edge is removed, it is more likely to be an intracommunity edge, by chance.

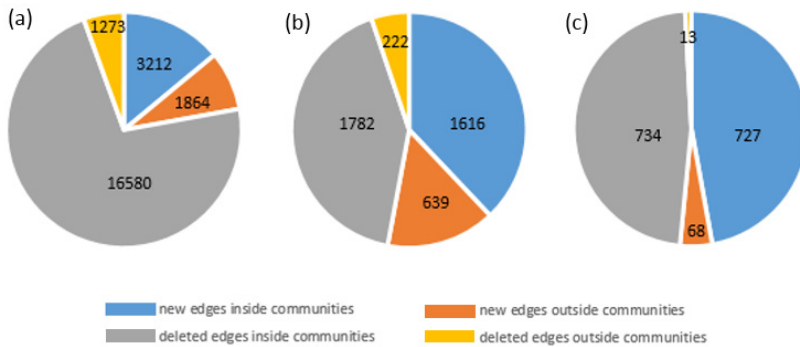


Fig. 3: . Distribution of new and deleted edges of the three networks collected in both phases. (a) Beauty; (b) Comic; (c) Graphics.

3.2 New edges of random case

To evaluate the numbers of new and deleted edges correctly, the actual numbers must be compared with a null model which adds or deletes edges randomly. If $G1 = (V1, E1)$ and $G2 = (V2, E2)$ are the networks of the first and second phase respectively, we randomly generate a new edge $\{u, v\}$ where $u \in V1 \cap V2, v \in V1 \cap V2$ and $\{u, v\} \notin E1$. This means that we connect a randomly chosen pair of vertices that existed in both June and August but were not linked by an edge in June.

Based on this strategy, for every network, the total number of edges added is equal to the number in the corresponding real network. Table 3 shows the average number, largest number, and smallest number of new *intracommunity* edges in all three networks after generating the randomly grown network 100 times. Taking the

Graphics network as an example, there should be 795 new edges, of which 727 are intracommunity (calculated from Fig. 3). From this table, the average number of intracommunity new edges in the random case is 297 which is much less than the real result, which is 727 edges. Even the largest value found, 329, is still much less than 727. For the other two networks, the results are consistent with the Graphics network. This answers our question: new edges occur inside communities more often than expected by chance.

Table 3: Intracommunity new edges of the random case in the three networks.

Network	Maximum	Minimum	Average	Real
Beauty	704	584	637	3212
Comic	652	548	597	1616
Graphics	329	254	297	727

Fig. 4 shows the distribution of the number of intracommunity edges added in each of the random networks. The star in each chart represents the number of intracommunity edges in the corresponding real network, which is always much greater than the numbers achieved in the random case. This allows us to reject the null hypothesis that the result is by chance. In principle, we could plot these curves analytically and calculate the extremely small probability that the real result could happen by chance, but we have not done so here.

3.3 Deleted edges of random case

Fig. 3 shows that most of the deleted edges are intracommunity edges, but this is to be expected because there are relatively few intercommunity edges to delete. We need to investigate whether deleted edges are more likely to be intercommunity than expected. We do this by simulating another shrunk network based on the original network. The strategy is to remove edges from this network randomly.

If networks $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are the networks of the first phase and second phase respectively, we randomly choose a edge $\{u, v\}$ where $u \in V_1 \cap V_2, v \in V_1 \cap V_2$ and $\{u, v\} \in E_1$. This means that we randomly choose a pair of vertices that existed in both June and August and were linked by an edge in June, and delete that edge.

Fig. 3 shows the number of deleted edges in the three networks. For these three networks, 17853, 2004, and 747 edges were removed, respectively.

In order to test the hypothesis that intercommunity edges are more likely to be deleted, we compare the number of deleted edges of the random case with the real network, in Fig. 5. Taking the Comic network (Fig. 5(b)) as an example, the average number of intercommunity deleted edges is around 75 and even the maximum, 97, is

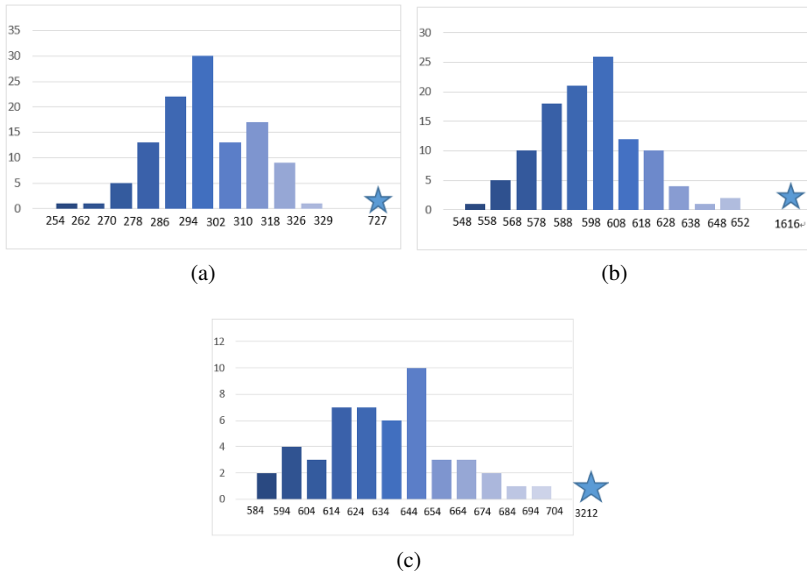


Fig. 4: Number of new intracommunity edges added in each network. (a) Graphics network (run 112 times); (b) Comic network (run 110 times); (c) Beauty network (run 50 times).

far less than the real result, 222. These results are less pronounced than for added edges (Section 3.2) but still show that intercommunity edge deletion is more common than expected by chance.

3.4 Biased network

Section 3.2 showed that new intracommunity edges are added far more often than could happen by chance, but a more interesting question is how much more often.

In order to measure this, we imagine a biased random agent that repeatedly adds new edges: each edge has a probability p to be an intracommunity edge; otherwise it is an intercommunity edge. We adjust the probability p until the number of intracommunity edges added is close to the real value. After testing several times, the probability values found for the three networks are 0.7 (Beauty), 0.75 (Comic), and 0.82 (Graphics). From Table 3 and Fig. 3, we can compute equivalent probabilities for an unbiased random agent: 0.12, 0.26, and 0.37, respectively. This means that, in the Beauty network for example, intracommunity edges are nearly six times more likely to be added than expected by chance.

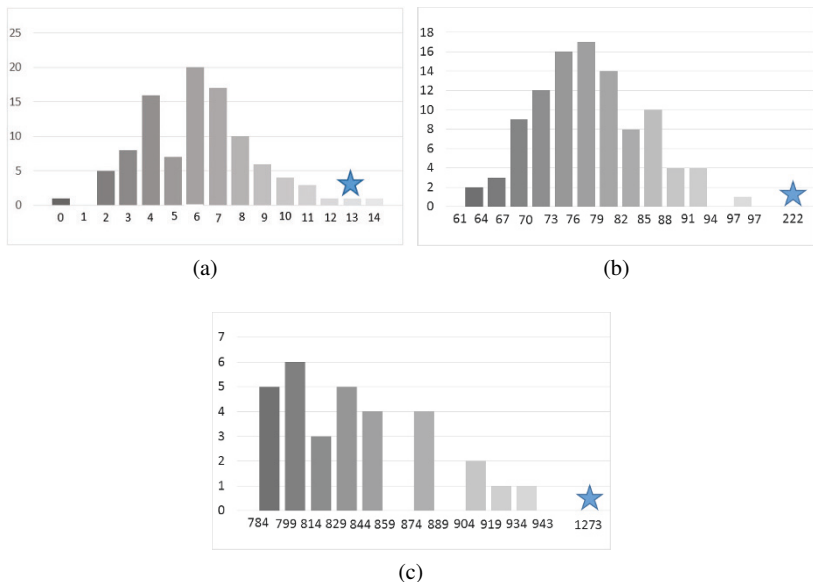


Fig. 5: Number of deleted intercommunity edges in each network. (a) Graphics network (run 100 times); (b) Comic network (run 100 times); (b) Beauty network (run 31 times).

4 Conclusions

We have shown that, at least for three network samples, the community structure of the Twitter follows network seems to become stronger as time passes, increasing the separation between communities.

It is important to emphasize that we have analysed only the network topology and not the details of the users or their tweets, which are outside the scope of this work. Therefore, we have no evidence of whether a community (in our sense) represents a single topic or viewpoint, or whether different communities represent opposing viewpoints. Indeed, because we only detect *disjoint* communities, it is unlikely that each community detected discusses only a single topic. Nevertheless, in cases where communities do correspond to viewpoints, this separation can be interpreted as polarization.

Our specific findings are:

1. New edges are intracommunity edges much more often than expected.
2. Deleted edges are intercommunity edges much more than expected.
3. When adding edges, users are about 3-4 times more likely to add an intracommunity edge than an intercommunity edge.

These observations probably underestimate the true effect. Because we collect small samples of the network, community detection is certain to be imperfect because some communities are split between the sample and the rest of the network and cannot be found. In the extreme case, if random communities were found, our results would be no different from the random null model with which we compare. If we had time to collect larger samples, we would therefore expect an even more pronounced effect. This is a good topic for future work.

It is interesting to speculate on the reason for the effect we observe. One possible explanation is the recommender system of Twitter: users receive suggestions about users that they might want to follow, and these are often users who are already in the same network community. Further work would be needed to find out whether the generation of new edges is consistent with Twitters recommendations (which are not revealed except to the users themselves). In any case, the recommender system cannot be the only explanation because of (2) above: Twitter never recommends users to unfollow. It seems more likely that users start following others after discovering them through the network structure itself; e.g., by retweets. New users (those that exist in the later snapshot but not the first) might even play a role in introducing existing users to each other and causing an edge to appear, even though we exclude these new users from our network samples.

Finally, it may be argued that, even if Twitter communities become more polarized over time, this might not be caused by the platform itself. The Twitter network may be converging over time to an underlying real-world network which is already highly polarized. Even so, Twitter provides the mechanisms to reflect and enhance this polarization, unlike traditional media and communication methods, which might tend to reduce it.

Future work

Further work is needed to estimate the probability with which a biased random agent chooses an intercommunity edge to delete. We have used a simple null model for our unbiased random agent, whereby vertices to connect are chosen uniformly randomly from all vertices in the sample. Numerous other null models are possible; for example, the agent might preferentially connect to popular (high-degree) users or to users with a similar name or description. In future, it would be useful to test other null models to rule out other possible explanations for the results found.

Another area of future work is to repeat the analysis with different community detection algorithms instead of Infomap. This is simple to do because we have kept the sampling and analysis phases separate, which would not be the case if we had used (e.g.) a local modularity [4] method to collect the network samples. A more challenging task would be to detect overlapping communities, instead of disjoint communities, in the networks. Overlapping communities are more realistic because many Twitter users have more than one interest and hence belong to multiple

communities. However, overlapping community detection is more difficult and the results would be harder to analyse.

Acknowledgements We are grateful to the anonymous referees for insightful comments that have improved the final version of the paper, especially its conclusions.

References

- [1] Adamic, L.A., Glance, N.: The political blogosphere and the 2004 us election: divided they blog. In: Proceedings of the 3rd international workshop on Link discovery, pp. 36–43. ACM (2005)
- [2] Amor, B., Vuik, S., Callahan, R., Darzi, A., Yaliraki, S.N., Barahona, M.: Community detection and role identification in directed networks: understanding the twitter network of the care.data debate. In: N. Adams N Heard (ed.) Dynamic networks and cyber-security., vol. abs/1508.03165. World Scientific Press (2016)
- [3] Cebrian, M., Rahwan, I., Pentland, A.S.: Beyond viral. *Commun. ACM* **59**(4), 36–39 (2016). DOI 10.1145/2818992. URL <http://doi.acm.org/10.1145/2818992>
- [4] Clauset, A.: Finding local community structure in networks. *Physical review E* **72**(2), 026,132 (2005)
- [5] Conover, M., Ratkiewicz, J., Francisco, M.R., Gonçalves, B., Menczer, F., Flammini, A.: Political polarization on twitter. *ICWSM* **133**, 89–96 (2011)
- [6] Dewey, C.: How twitter makes the political echo chamber worse. <https://www.washingtonpost.com/news/the-fix/wp/2013/09/03/how-twitter-makes-the-political-echo-chamber-worse/> (2013)
- [7] DiFonzo, N.: The echo-chamber effect. <http://www.nytimes.com/roomfordebate/2011/04/21/barack-obama-and-the-psychology-of-the-birther-myth/the-echo-chamber-effect> (2011)
- [8] Fortunato, S.: Community detection in graphs. *Physics reports* **486**(3), 75–174 (2010)
- [9] Newman, M.E.: Mixing patterns in networks. *Physical Review E* **67**(2), 026,126 (2003)
- [10] Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *Physical review E* **69**(2), 026,113 (2004)
- [11] Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* **105**(4), 1118–1123 (2008)

Semantic Stability in Wikipedia

Darko Stanisavljevic, Ilire Hasani-Mavriqi, Elisabeth Lex, Markus Strohmaier and Denis Helic

Abstract In this paper we assess the semantic stability of Wikipedia by investigating the dynamics of Wikipedia articles' revisions over time. In a semantically stable system, articles are infrequently edited, whereas in unstable systems, article content changes more frequently. In other words, in a stable system, the Wikipedia community has reached consensus on the majority of articles. In our work, we measure semantic stability using the Rank Biased Overlap method. To that end, we preprocess Wikipedia dumps to obtain a sequence of plain-text article revisions, whereas each revision is represented as a TF-IDF vector. To measure the similarity between consequent article revisions, we calculate Rank Biased Overlap on subsequent term vectors. We evaluate our approach on 10 Wikipedia language editions including the five largest language editions as well as five randomly selected small language editions. Our experimental results reveal that even in policy driven collaboration networks such as Wikipedia, semantic stability can be achieved. However, there are differences on the velocity of the semantic stability process between small and large Wikipedia editions. Small editions exhibit faster and higher semantic stability than large ones. In particular, in large Wikipedia editions, a higher number of successive revisions is needed in order to reach a certain semantic stability level, whereas, in small Wikipedia editions, the number of needed successive revisions is much lower for the same level of semantic stability.

Darko Stanisavljevic (e-mail: darko.stanisavljevic@v2c2.at)*✉
VIRTUAL VEHICLE Research Center, Inffeldgasse 21a Graz Austria,

Ilire Hasani-Mavriqi (e-mail: ihasani@know-center.at)*✉
Graz University of Technology and Know-Center GmbH, Inffeldgasse 13, Graz, Austria,

Elisabeth Lex (e-mail: elisabeth.lex@tugraz.at)✉ · Denis Helic (e-mail: dhelic@tugraz.at)✉
Graz University of Technology, Inffeldgasse 13, Graz, Austria

Markus Strohmaier
GESIS and University of Koblenz-Landau, Unter Sachsenhausen 6-8, Cologne, Germany, e-mail: markus.strohmaier@gesis.org

*Both authors contributed equally to this work.

Key words: semantic stability, semantic similarity, TF-IDF, RBO, Wikipedia

1 Introduction

Wikipedia is one of the largest, freely accessible web-based encyclopedias and its content is open for editing by users. Wikipedia articles are mainly a contribution of volunteer editors who collaboratively create and manage the largest repository of human knowledge. This way, different editors can contribute with their expertise, ideas and opinions. Wikipedia contributors, however, may have different motivations and opinions, for example, it may take some time for them to agree if sufficient and correct information is provided within an article. If editors have different point of views on a particular topic, especially on controversial topics, they might end up overwriting each others content such that articles cannot become semantically stable. These are also known as edit wars [3, 5, 13, 17]. On the contrary, if Wikipedia editors achieve consensus on the content, implicitly, articles become semantically stable.

Problem & objectives. The goal of this paper is to investigate the semantic stability process in collaboration networks, such as Wikipedia, that are driven based on policies, guidelines and community standards. Based on these policies, both editors' behavior and the process of article production is managed [7].

Approach & methodology. In order to assess the semantic stability of Wikipedia, we turn to semantic similarity of consecutive revisions of Wikipedia articles. Semantic similarity of two textual documents expresses the extent to which two documents deal with semantically similar topics or content. This concept is key to understanding the comparison of documents written in natural language. Typically, semantic similarity is calculated by means of document statistics. An advantage of statistical approach is that it does not require predefined models, which describe the meaning of particular words (terms). The method applied in this work, i.e., Rank Biased Overlap, is also a statistical method and it is first introduced in [16]. The basic procedure carried out during the calculation of the semantic similarity is the modeling of the semantic space in accordance with the term distribution in a corpus of documents. In such a space, each document is represented by a vector and semantic similarity is calculated by performing vector operations on those vectors. This approach is based on the distributional hypothesis, according to which the terms with similar meanings show tendency to appear in similar contexts [8].

The concept of semantic stability applied in our paper is based on the work presented in [15], which studies the semantic stability of social tagging systems. In our work, we are interested in the semantic stability of Wikipedia. Thus, we take a Wikipedia corpus of documents that contains the complete edit history for each article and which includes all existing article revisions. The following Wikipedia language editions are used: English, German, French, Spanish, Italian, Czech, Finnish (Suomi), Danish, Greek and Swedish. The intention behind the choice of these particular languages is to have five Wikipedia editions with a large number of articles and five smaller editions. This enables us to study the relation between semantic stability and

corpus size. Our long term goal is to investigate the consensus building process in Wikipedia based on the semantic stability. Authors of [15] state that semantic stability implies implicit consensus on the description of a resource in a social tagging system.

Findings & contributions. One of the contributions of our work is the software solution that we provide as an open source project¹, which is highly modular, configurable and flexible and can be applied by anyone looking for an efficient way to analyze the semantics of natural language documents contained, for example, in the Wikipedia XML dump files. From the empirical point of view, we conduct experiments in 10 different Wikipedia language editions and discuss the experimental results and their implications. Our experimental results reveal that the mean semantic stability of large Wikipedia editions is significantly lower compared to the mean semantic stability of small Wikipedia editions. In particular, in large Wikipedia editions, a higher number of successive revisions is needed in order to reach a certain semantic stability level, whereas, in small Wikipedia editions for the same level of semantic stability, the number of successive revisions needed, is much lower.

2 Technical Approach

2.1 Preliminaries

Particularly important for this paper is the theory describing: (i) evaluation of importance of terms in a single document or in a corpus of documents and their representation in a form of matrix - TF-IDF (Term Frequency - Inverse Document Frequency), (ii) calculation of semantic similarity measure and (iii) calculation of semantic stability over time.

We represent each revision of the parsed Wikipedia articles as a TF-IDF vector. *Term Frequency - Inverse Document Frequency* is one of the methods in the theory of Information Search and Retrieval used to represent the relevance of terms in a document belonging to a collection of documents - *corpus* [2, 9, 14, 18].

The comparison of the TF-IDF vectors is performed using a modified version of RBO (Rank Biased Overlap) method as in [15]. However, our approach is flexible and can be extended to include additional similarity measures. The RBO method is used to calculate the similarity measure of two given vectors, each of them representing the rankings of terms contained in a single Wikipedia article. Its main characteristic is that it takes the cumulative overlap of the given rankings as a measure for similarity. It is represented with the following mathematical equation:

$$RBO(\sigma_1, \sigma_2, p) = (1 - p) \sum_{d=1}^{\infty} \frac{2 * \sigma_{1:l:d} \cap \sigma_{2:l:d}}{|\sigma_{1:l:d} + \sigma_{2:l:d}|} p^{(d-1)} \tag{1}$$

where σ_1 and σ_2 are not necessarily conjoint lists of ranking and $\sigma_{1:l:d}$ and $\sigma_{2:l:d}$ are ranked lists at depth d . RBO evaluates to a value in the range $[0, 1]$, where 0 means disjoint and 1 means identical. The parameter p defines the steepness of the weights

¹ <https://doi.org/10.5281/zenodo.153891>

and takes a value in interval $(0 \leq p < 1)$. When $p = 0$, RBO considers only the top ranked item of the lists and its value is either 0 or 1. When p is arbitrarily close to 1 the weights are almost the same for all depths and the analysis is arbitrarily deep.

The similarity measure described in Equation 1 is used as basis for determining the semantic stability over time. Based on [15], for a given value of RBO threshold k , an article is semantically stable if its RBO value at the point of time t is equal or higher than the threshold k . A rather simple mathematical formulation of this method for inspection of stabilization process in a given data set is as following:

$$f(t, k) = \frac{1}{n} \sum_{t=1}^n \begin{cases} 1, & \text{if } RBO(\sigma_{t-1}, \sigma_t, p) \geq k \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Based on the Equation 2, for each article in a Wikipedia corpus, the rank-biased overlap similarity measure is calculated. Inputs are the revisions before and after the time point t as well as the parameter p . If the calculated similarity is equal or greater than the threshold k , 1 is added to the sum, otherwise 0 is added. With no more articles in corpus to iterate, the sum is divided by the total number of iterated articles from the Wikipedia corpus. Thus, the result will be the percentage of the stable articles at time-point t for a predefined threshold value k .

For our experiments, the rank-biased overlap similarity measure algorithm is parametrized with the $p = 0.9$ which means that the first ten ranks of the ranking list have 86% of the weight of the evaluation as stated in [15]. Empirically, we also find that $p = 0.9$ is appropriate because of the value of parameter d (depth of evaluation) chosen for rank-biased overlap. This means that the TF-IDF vectors will be checked for similarity only up to the depth of 20. Of course, one can take a much higher depth, but that will increase the computation time as well as the storage space. Namely, the TF-IDF vector representing a single revision of an arbitrary article can have several thousands of values, but not all of those values are stored. Only the values up to the depth needed for rank-biased overlap calculation are stored. So, if 20 elements are used for rank-biased overlap measure, the first 10 elements of the ranking weight 86% of the evaluation and the other 10 elements weight only 14%. It is exactly because of this fact that there is no need to do the similarity calculation for much higher depths as those are not regarded as very important. In every case, the top 20 (most-weighted) elements of the TF-IDF vector are more than enough to precisely describe the semantics of the article revision they represent.

2.2 Experimental Setup

We study two different aspects of the stabilization process: (i) semantic stabilization of the Wikipedia corpus over a predefined period of time and (ii) semantic stabilization of the Wikipedia corpus after a number of successive revisions. The idea behind the examination of the Wikipedia corpus stabilization over the time is to choose a point in time t and count the number of articles existing at that point in time and

the number of articles existing at that point in time that are also semantically stable. This is possible because of the fact that every article revision is uniquely identified in the database by the compound key consisting of the article ID and the revision timestamp.

Another way to inspect the stabilization process of the document corpus is to find out how many successive revisions are required before a percentage of the available articles becomes stable (in reference to the stability threshold). The idea is very similar to the previously discussed one, but now it is assumed that all articles have the first revisions starting at the same date and time. The timestamp information is now completely neglected and only the number of revisions per article is important. So, at the beginning, the first value of the similarity vectors of all articles is examined. The stability threshold takes the maximal value at the beginning of the calculation, 1. If the desired percentage of the articles is stable, the next value of the similarity vector is inspected. If not, the threshold is decreased and the calculation is repeated until the value of the stability threshold, for which the desired percentage of articles is stable, is found. Analysing the semantic stability from two different point of views, provides more useful insights about the examined corpus.

Dataset Preprocessing. The Wikimedia² provides XML dumps of all active Wikipedia projects. The basic building block of all Wikipedia editions is a page. Every page represents an article and every article has at least one, but usually more than one, revision. There are articles in bigger Wikipedia editions which have tens of thousands of revisions.

We analyze 10 Wikipedia language editions, five of which are (randomly selected) small language editions and the remaining five are the largest language editions. Our goal is not to analyze the full Wikipedia corpus of the large editions, thus, the sampled data of 10 thousand randomly selected articles with their complete revision history is used for 8 out of 10 Wikipedia editions. Only Czech and Finnish Wikipedia corpus is fully analyzed.

3 Results and Discussion

Figure 1 compares the stabilization process between small and large Wikipedia language editions over a period of time. A portion of the stable articles (in percentages) is shown for a chosen point in time t , in order to spot periods of increased stability or instability of an article corpus. The plots in Figure 1 correspond to the RBO threshold $k = 0.8$. We run experiments with two other values: $k = 0.4$ and $k = 0.6$, to investigate the role of the threshold parameter k in the stability calculation method proposed in [15]. Once the similarities of all revisions of a single Wikipedia article are calculated, the value representing the similarity in a given moment of time t is taken and compared to the value of the parameter k . Our intuitive assumption is that, for a low value of RBO threshold k , there are a lot of articles in the examined corpus, whose stability value in a given instant of time is higher than the chosen threshold.

² <https://dumps.wikimedia.org/>

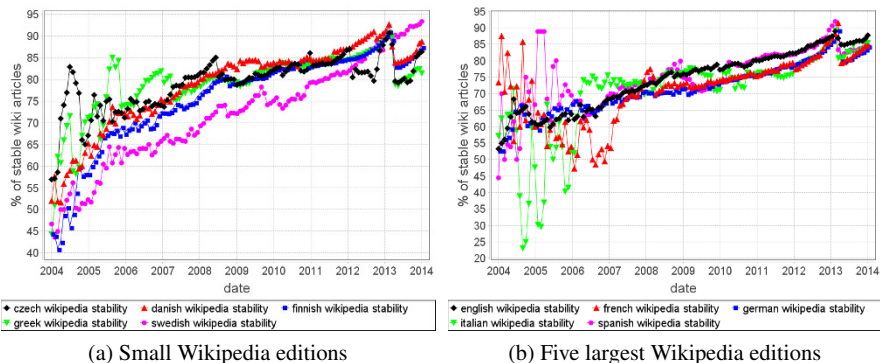


Fig. 1: Semantic stabilization of the Wikipedia corpus over a period of time. Percentages of stable articles (y-axis) are shown in relation to a predefined period of time (x-axis) for (a) small and (b) large Wikipedia editions. Semantic stability curves shown, correspond to the RBO threshold $k = 0.8$ and steepness parameter $p = 0.9$. For illustration, consider the plot in (a), for a chosen point in time, (e.g.,) year 2008, in (e.g.,) Czech edition, is indicated that 70% of articles have reached a semantic stability equal or higher than 0.8. The steepness of the stabilization curves remains the same over different parameters k , however, the percentage of stable articles decreases with increasing k . Comparing plots in (a) and (b), one can see that the mean semantic stability of small Wikipedia editions is significantly higher in contrast to large ones. This is in line with the fact that small Wikipedia editions contain large portions of articles simply translated from the English Wikipedia, for example. Such articles are usually rarely changed substantially and they increase the overall stability of small editions. In contrary, the editorial process in large editions is much more dynamic.

Our results are consistent with our initial assumptions. Thus, as the value of the RBO threshold increases, the number of stable articles decreases. The document corpus stability is inversely proportional to the value of parameter k . However, the steepness of the stabilization curves remains the same over different parameters k , thus, we include plots for only $k = 0.8$ to show the least stability.

From the plot in Figure 1a, it is noticeable that all small Wikipedia editions exhibit semantic stability variations in almost the same range (with a deviation $\pm 2\%$ from the average). The only exception to this is the case of Swedish Wikipedia that has the semantic stability well below the average semantic stability of the other four small Wikipedia editions.

Figure 1b shows that in large Wikipedia editions, semantic stabilization curves oscillate more at the beginning of the editorial process compared to small editions. Thus, they are, on average, more unstable than the small Wikipedia editions. Our explanation for this is that the small Wikipedia editions consist mainly of articles which are the translated versions of the articles from the main Wikipedia editions (for example from the English Wikipedia). Once translated and created, such articles

are rarely edited a lot. Whereas, in large editions such as in the English one, a higher number of new articles that are authored from scratch is present. Of course, the editorial process of such articles is more dynamic.

We observe a very interesting phenomenon in both plots in Figure 1, namely, in both small and large Wikipedia editions, a sudden increase of the semantic stability is noted, with a peak around year 2013. Right after this point of time, the stability decreases for all Wikipedia editions and then continues to increase again. We wanted to find an explanation for this observation by contacting the Wikipedia community by writing several posts in the *Wikimedia.org*³ mailing list, but we did not receive any plausible answer. Some of the assumptions are that: some of the Wikipedia servers were down for a short maintenance, or some of the Wikipedia maintenance bots were active and editing Wikipedia contents was shortly blocked or malfunctioning of Wikipedia servers was induced by malicious software or hacker attacks. But, the temporary peak in semantic stability in year 2013 could also be seen as a consequence of a change in Wikipedia policies of how to handle edit wars (e.g, the introduction of a new rule such as the three-revert rule). Still, no hard evidence was brought into light.

Figure 2 visualizes the number of consecutive revisions per article needed to achieve the stability of 95% in both small and large Wikipedia editions. This means that 95% of articles in a corpus become semantically stable, evaluated based on different RBO (for $p = 0.9$) thresholds k (y-axis in Figure 2), after r consecutive revisions (x-axis).

In Figure 2a, 95% of stable articles is reached after, for example, 70 revisions for the Greek Wikipedia and 30 or less revisions for all other small Wikipedia editions. It can be seen that for the Greek Wikipedia, 95% of the articles has the stability of 0.5 or higher after almost 35 revisions, where $k = 0.5$ is considered as a medium stability [15]. From this fact one can conclude that the Greek Wikipedia edition is the most frequently edited one amongst the analyzed small editions. The Czech and Swedish editions are showing much more semantic stability, 95% of the article corpus of this two editions has the semantic stability of 0.5 or higher after only about 5 revisions.

Figure 2b shows the stabilization process of large Wikipedia editions where the achieved stability is 95%. This time, as expected, the English Wikipedia is the most unstable one. Almost the complete corpus of analyzed articles becomes stable after almost 95 revisions of each article. The medium semantic stability of the corpus that is defined by the value of parameter $k = 0.5$ is, in the case of English Wikipedia, reached after about 45 revisions, and in the case of the French one (the most stable one) after about 30 revisions.

These results are in line with the fact that larger communities contribute to the largest Wikipedia editions (e.g., English, German or French), in comparison to the communities editing the small Wikipedia editions, written in languages, which are only used by a very small percent of the world population. Large authoring community indicates a heterogeneous community based on authors' expertise, ideas and opinions, which in turn implies that the contributed content is more colorful. If

³ <https://lists.wikimedia.org/mailman/listinfo/wiki-research-1>

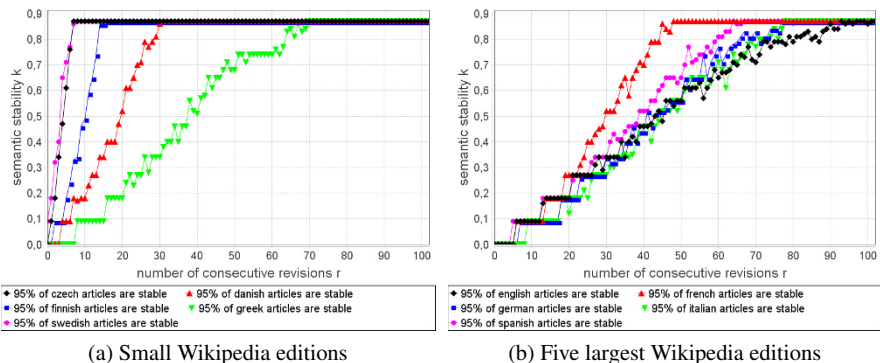


Fig. 2: Semantic stabilization of the Wikipedia corpus after a number of successive revisions. 95% of articles in a corpus become semantically stable, evaluated based on different RBO thresholds k (y-axis), after r consecutive revisions (x-axis). The plot in (a) illustrates that almost all small editions exhibit, at the beginning, a fast increase of the stabilization curves, which remain relatively stable after few successive revisions. An exception presents the Greek edition, which is the most frequently edited among the small ones. The plot in (b) depicts that the stabilization process in large editions is delayed. This indicates that in large editions a higher number of successive revisions is needed in order to reach the same semantic stability level as in small Wikipedia editions. These results are consistent with the fact that the size of the community contributing to the large editions, such as English, can not be compared to the small ones. Large communities are characterized with heterogeneous contributors’ expertise, motivation and opinions, which implicates that it takes time until contributors agree if sufficient and correct information is provided within an article.

content contributors have different point of views on a particular topic, especially on controversial topics, they might end up overwriting each others content such that articles cannot become semantically stable. Thus, in large Wikipedia editions a higher number of revisions is needed until contributors agree if sufficient and correct information is provided within an article.

Key findings. Our findings can be summarized as follows: even in policy driven collaboration networks such as Wikipedia, semantic stability can be achieved. However, there are differences on the velocity of the semantic stability process between small and large Wikipedia editions. In large Wikipedia editions, semantic stability curves oscillate more at the beginning of the editorial process compared to small editions. Thus, the mean semantic stability of large Wikipedia editions is significantly lower in contrast to small Wikipedia editions. In other words, small Wikipedia editions stabilize faster and achieve higher levels of semantic stability.

4 Related work

The process of consensus reaching among Wikipedia editors has been on the focus of many recent studies [1, 3, 5, 6, 7, 13, 17]. Authors in [5] study the problem of edit wars in Wikipedia and model this phenomenon using agent-based systems, based on theories of group stability and reinforcement learning. Authors show that consensus is reached faster if the number of credible or trustworthy agents and agents with a neutral point of view is increased. In the contrary, consensus is hindered when agents with opposing views are in equal proportion. Similarly, authors in [13] apply also an agent-based model to emulate conflict scenarios in edit wars and validate their model by empirical Wikipedia data. Recently published work [3] uses hidden Markov models to approximate and characterize the computational structure of conflicts in Wikipedia.

The work presented in [7] investigates the role of conflict in the editorial process in Wikipedia by studying talk pages. Experimental results reveal that conflict is central to the editorial processes of Wikipedia; it is a generative friction that is used by Wikipedia editors as part of a coordinated effort within the community to improve the quality of articles.

There are several research approaches published in the field of semantic similarity measurements [4, 10, 11, 12]. Hajian et. al. [4] propose a multi-tree similarity algorithm as a non-linear technique for measuring similarity based on hierarchical relations which exist between attributes of entities in an ontology. This method compensates for the lack of semantic relatedness among features using taxonomic relations that exist among the features of two entities. In [10] authors implement a probabilistic method of measuring semantic similarity for real-world noisy short texts like microblog posts. Their method adds related Wikipedia entities to a short text as its semantic representation and uses the vector of entities for computing semantic similarity. The work presented in [11] shows that the combination of knowledge and corpus-based word-to-word similarity measures can produce higher agreement with human judgment than any of the individual measures. Authors in [12] present an approach for measuring semantic similarity between words using the snippets returned by Wikipedia and the five different similarity measures of association. Their results demonstrate that the snippets in Wikipedia have a significant influence on the accuracy of semantic similarity measure between words.

The Rank Biased Overlap or shortly RBO method is introduced in [16]. Our study is based on the scientific work [15], in which a modified version of RBO is applied to investigate the semantic stability of social tagging systems. However, in our work we assess the semantic stability of Wikipedia articles.

5 Conclusion and Future Work

In this work, we study the semantic stabilization of Wikipedia with a focus on the dynamics of Wikipedia articles' revisions over time. Our experimental results reveal that: (i) the analyzed Wikipedia language editions show medium semantic stability

and (ii) large Wikipedia editions exhibit a significantly lower mean semantic stability value compared to the small Wikipedia editions.

Our first findings are in line with the research results of the work presented in [15], in which authors state that natural languages are semantically stable in their nature. In our case, all the analyzed datasets have at least medium semantic stability.

Our second experimental results indicate that the large Wikipedia editions, which were utilized for the purpose of this paper are semantically less stable than the small ones. This observation can be logically explained by the fact that large Wikipedia editions have much more contributors than the small ones. The sheer size of the community supporting and developing the English Wikipedia edition cannot be compared to e.g., the size of community working on the Czech Wikipedia edition. Having many more users contributing to the content means that higher semantic instability is brought to the system. The users of English Wikipedia are changing the content of the articles much more than the users of small Wikipedia editions. Additionally, many articles available in small Wikipedia editions are simply translations of the articles found in the English Wikipedia. Once translated, such articles are rarely changed significantly, which contributes to a higher semantic stability of the small Wikipedia editions.

One of the limitations of our work is that we evaluated only sampled data for the large Wikipedia editions. However, our software solution is flexible and could be easily extended to analyze the full Wikipedia corpus of the large editions.

For future work, we plan to investigate the consensus building among editors in different Wikipedia categories, in order to find out if there are categories that are unstable. We also want to specifically study the semantic stability of articles marked as controversial. One of our future plans is to combine the content based approach introduced in this work with a network based approach. Vandalism detection is also a topic that could benefit from our work.

Acknowledgements This work is supported by the Know-Center, the VIRTUAL VEHICLE Research Center and the AFEL project funded from the European Union's Horizon 2020 research and innovation programme under grant agreement No 687916. The Know-Center and the VIRTUAL VEHICLE Research Center are funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - under the auspices of the Austrian Ministry of Transport, Innovation and Technology, the Austrian Ministry of Economics and Labor and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency (FFG).

References

- [1] Biancani, S.: Measuring the Quality of Edits to Wikipedia. In: Proceedings of The International Symposium on Open Collaboration, OpenSym '14. ACM, New York, NY, USA (2014)
- [2] Debole, F., Sebastiani, F.: Supervised Term Weighting for Automated Text Categorization. In: Proceedings of the 2003 ACM Symposium on Applied Computing, SAC '03. ACM, New York, NY, USA (2003)
- [3] DeDeo, S.: Conflict and computation on wikipedia: A finite-state machine analysis of editor interactions. *Future Internet* **8**(3) (2016)

- [4] Hajian, B., White, T.: Measuring Semantic Similarity using a Multi-tree Model. In: Proceedings of the 9th Workshop on Intelligent Techniques for Web Personalization and Recommender Systems, ITWP 2011. CEUR Workshop Proceedings (2011)
- [5] Kalyanasundaram, A., Wei, W., Carley, K.M., Herbsleb, J.D.: An Agent-based Model of Edit Wars in Wikipedia: How and when is Consensus Reached. In: Proceedings of the 2015 Winter Simulation Conference, WSC '15. IEEE Press, Piscataway, NJ, USA (2015)
- [6] Müller-Birn, C., Dobusch, L., Herbsleb, J.D.: Work-to-rule: The Emergence of Algorithmic Governance in Wikipedia. In: Proceedings of the 6th International Conference on Communities and Technologies, C&T '13. ACM, New York, NY, USA (2013)
- [7] Osman, K.: The Role of Conflict in Determining Consensus on Quality in Wikipedia Articles. In: Proceedings of the 9th International Symposium on Open Collaboration, WikiSym '13. ACM, New York, NY, USA (2013)
- [8] Sahlgren, M.: An Introduction to Random Indexing. In: Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE '05. Copenhagen, Denmark (2005)
- [9] Salton, G., Buckley, C.: Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management: an International Journal* **24**(5) (1988)
- [10] Shirakawa, M., Nakayama, K., Hara, T., Nishio, S.: Probabilistic semantic similarity measurements for noisy short texts using Wikipedia entities. In: Proceedings of the 22nd ACM international conference on Conference on information & knowledge management, CIKM '13. ACM, New York, NY, USA (2013)
- [11] Stefanescu, D., Rus, V., Niraula, N.B., Banjade, R.: Combining Knowledge and Corpus-based Measures for Word-to-Word Similarity. In: Proceedings of the Twenty-Seventh International Florida Artificial Intelligence Research Society Conference, FLAIRS '14. AAAI Press, Palo Alto, California (2014)
- [12] Takale, S.A., Nandgaonkar, S.S.: Measuring Semantic Similarity between Words Using Web Documents. *International Journal of Advanced Computer Science and Applications, IJACSA* **1**(4) (2010)
- [13] Török, J., Iñiguez, G., Yasseri, T., San Miguel, M., Kaski, K., Kertész, J.: Opinions, conflicts, and consensus: Modeling social dynamics in a collaborative environment. *Phys. Rev. Lett.* **110** (2013)
- [14] Turney, P.D., Pantel, P.: From Frequency to Meaning Vector Space Models of Semantics. *Journal of Artificial Intelligence Research* **37**(1) (2010)
- [15] Wagner, C., Singer, P., Strohmaier, M., Huberman, B.A.: Semantic Stability in Social Tagging Streams. In: Proceedings of the 23rd International Conference on World Wide Web, WWW '14. ACM, New York, NY, USA (2014)
- [16] Webber, W., Moffat, A., Zobel, J.: A similarity Measure for Indefinite Rankings. *ACM Transactions on Information Systems, TOIS* **28**(4) (2010)
- [17] Yasseri, T., Kertész, J.: Value production in a collaborative environment. *Journal of Statistical Physics* **151**(3) (2013)
- [18] Zaman, A.: Stop Word Lists in Document Retrieval Using Latent Semantic Indexing: an Evaluation. *Journal of E-Technology* **3**(1) (2012)

Coopetition and Cooperosity over Opinion Dynamics

Domenico Tangredi, Raffaele Iervolino and Francesco Vasca

Abstract In the heterogeneous Hegselmann–Krause (HK) opinion dynamics network, the existence of edges among the agents depend on different connectivity thresholds. A new version of this model is here presented, by using the notions of *coopetition* and *cooperosity*. Such concepts are defined by combining the representation of the cooperation, competition and generosity behaviours. The proposed HK model is recast as a piecewise linear system with the state space partitioned into convex polyhedra defined by the agents influence functions. A sufficient condition for the local asymptotic stability, i.e., the consensus, is formulated as a set of linear matrix inequalities whose solution provides a continuous piecewise quadratic Lyapunov function. Numerical results show the effectiveness of the proposed approach.

1 Introduction

In the last decades there has been a considerable growth of interest in the analysis of social networks from the scientific perspective of systems theory [15]. In the so-called Hegselmann–Krause (HK) model the dynamics of each agent is described by a scalar differential equation whose discontinuous right hand side depends on the differences between the agent state with the others [6, 12, 13]. In particular, the state value of the i -th agent, say ξ_i , is a measure of the intensity of its opinion or attitude toward a particular purpose or action [3]. The state interpretation as an agent's attitude is more appropriate for the analysis proposed in this paper, however the more common term opinion will be also used. The opinions difference between each pair of agents

Domenico Tangredi (e-mail: domenico.tangredi@unisannio.it)✉ · Francesco Vasca (e-mail: vasca@unisannio.it)✉

Department of Engineering, University of Sannio, Piazza Roma 21, 82100 Benevento, Italy

Raffaele Iervolino (e-mail: rafierv@unina.it)✉

Department of Electrical and Information Technology Engineering, University of Naples Federico II, 80125 Napoli, Italy

is weighted by the so-called *influence function* which is zero if the absolute value of such difference is larger than a given *connectivity threshold* [14, 24].

The use of different connectivity thresholds with influence functions depending on the sign of the attitudes difference, allows one to introduce the concepts of *coopetition* (cooperation and competition) and *cooperosity* (cooperation and generosity). The former concept has been widely analyzed in the literature. The term *coopetition* is a neologism introduced to represent an interaction between agents that compete and cooperate at the same time [23]. Coopetitive networks have been modelled as signed graphs where the positive and negative edges represent the cooperative and competitive interactions, respectively [7, 21]. A variation of the coopetitive model with sign invariant agents opinions has been proposed in [2]. In this paper we say that two agents i and j *cooperate* if both edges from i to j and vice-versa are active. Under cooperation, we call *coopetition* the behavior of i versus j when the agent i has a lower attitude ($\xi_i < \xi_j$) and the cooperation with j contributes to an increase of ξ_i . Analogously, the term *cooperosity* [22] is a neologism introduced by the authors to represent the generosity of the agent i who cooperates with the agent j when he has a better skill ($\xi_i > \xi_j$) and the agents cooperation results in a decrease of ξ_i . Following this interpretation, in the HK model it always happens that the *coopetition* of i versus j corresponds to the *cooperosity* of j versus i , and viceversa.

The analysis of the convergence to a consensus, i.e., all agents reach the same opinion, has been widely considered in the literature [1, 24]. If the connectivity thresholds of the agents are different, i.e., the network is heterogeneous, clusters or consensus are more sensitive to the agents initial opinions, also for the case of few agents [11, 18, 19, 20]. In this paper we reformulate the HK model in a piecewise linear (PWL) form and we propose a sufficient condition for the asymptotic stability to the origin, which is the equilibrium point corresponding to the consensus, by using a Lyapunov approach. The existence of a piecewise quadratic (PWQ) Lyapunov function is formulated in terms of linear matrix inequalities obtained by extending the approach adopted for conewise linear systems [10].

The rest of the paper is organized as follows. In Section 2 we present our opinion dynamics model with a more general *influence function* suitable for the analysis of the *coopetition* and *cooperosity* behaviours. In Section 3 the model is represented in PWL form and in Section 4 the stability problem of the consensus is tackled by using a PWQ Lyapunov function. The numerical simulations analyzed in Section 5 confirm the effectiveness of our approach. Section 6 concludes the paper.

2 Coopetition and cooperosity

v In this section the *coopetition* and *cooperosity* concepts are used to determine a new formulation of the HK model. The classical HK model consists of a set of N autonomous agents, whose attitudes are state variables $\xi_i \in [0, 1]$ whose dynamics are described by

$$\dot{\xi}_i = \sum_{j=1}^N \phi_{ij}(\xi_i, \xi_j)(\xi_j - \xi_i) \tag{1}$$

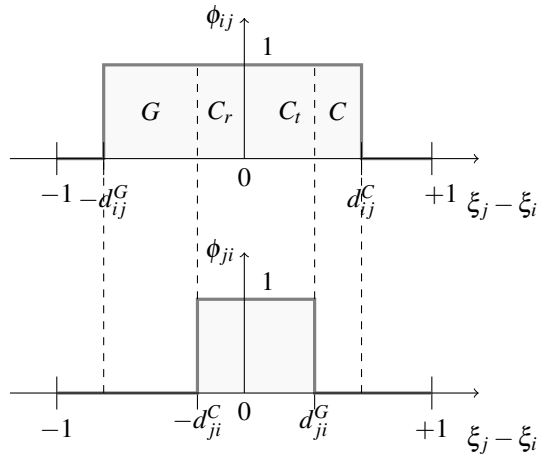
for $i = 1, \dots, N$, where for simplicity we omit the time dependence of the variables ξ_i . The *influence function* $\phi_{ij}(\xi_i, \xi_j) : [0, 1]^2 \rightarrow \{0, 1\}$ is equal to 1 when ξ_j influences the opinion evolution of the agent i , and 0 otherwise. For all agents, we propose an influence function that depends on the difference $\xi_j - \xi_i$ as follows

$$\phi_{ij}(\xi_i, \xi_j) = \begin{cases} 1, & \text{if } -d_{ij}^G \leq \xi_j - \xi_i \leq d_{ij}^C \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where the constant $d_{ij}^G \in [0, 1]$ is the connectivity threshold bounding the *generosity* of the agent i versus the agent j and the constant $d_{ij}^C \in [0, 1]$ is the connectivity threshold bounding the *competition* of the agent i versus the agent j . Without loss of generality we set $\phi_{ii} = 0$.

In order to explain the *cooperosity* and *coopetition* behaviours, let us consider Fig. 1. If $\phi_{ij} = 0$ the agent i is not connected to the agent j and the dynamics of ξ_i is not directly influenced by the state ξ_j , while the opposite is allowed, i.e., ϕ_{ji} can be equal to 1 although this scenario is not represented in Fig. 1.

Fig. 1 Possible influence functions ϕ_{ij} and ϕ_{ji} , both as a function of $\xi_j - \xi_i$, showing the five different behaviours of i vs. j which can occur depending on the relative opinions. In particular, when $-1 \leq \xi_j - \xi_i < -d_{ij}^G$ and $d_{ij}^C < \xi_j - \xi_i \leq +1$ the two agents do not influence each other. The areas identified by the letters G, C_r, C_l and C indicate the *generosity, cooperosity, cooperative* and *competitive* behaviours of i vs. j , respectively.



If $\phi_{ij} = 1$ it means that the dynamics of the agent i is influenced by the opinion of the agent j , while ϕ_{ji} can be either 0 or 1. Through the concepts of *coopetition* and *cooperosity* we can better specify the actions of i versus j when $\phi_{ij} = 1$. Let us consider the case $\xi_j < \xi_i$ with the influence functions in Fig. 1. From (1) it follows that the term $\xi_j - \xi_i$ contributes negatively to the derivative of ξ_i , i.e., it decreases the attitude of i . This identifies either *generosity* of i versus j if j is not influenced by

i (see the region G in Fig. 1), or *cooperosity* if the agent j is influenced by i (see the region C_r in Fig. 1).

The *cooperative* and *competitive* behaviours occur when $\xi_j > \xi_i$. Since $\xi_j - \xi_i$ is positive, from (1) it follows that the term $\xi_j - \xi_i$ contributes positively to the derivative of ξ_i . In other words, being the attitude of the agent j larger than the attitude of the agent i , the agent i competes with the agent j in the sense that i improves its attitude. This identifies either *cooperation* of i versus j if the agent j is influenced by i (see the region C_t in Fig. 1), or *competition* if the agent j is not influenced by i (see the region C in Fig. 1).

The different interactions over the network allow to define for each agent the corresponding benefits β_i and costs σ_i , $i = 1, \dots, N$. Then a fitness can be defined similarly to the one considered in [16] which extends the one adopted for the repeated Prisoner's Dilemma. In particular, we define the fitness of the agent i as the average of the benefits minus the costs, evaluated over the number of agents connected to i :

$$f_i = \frac{1}{\sum_{j=1}^N \phi_{ij}} (\beta_i - \sigma_i) \quad (3)$$

with

$$\beta_i = \sum_{j=1}^N \phi_{ij} \left[\phi_{ji} \beta_i^{C_t} + (1 - \phi_{ji}) \beta_i^C \right] \text{step}(\xi_j - \xi_i) \quad (4)$$

$$\sigma_i = \sum_{j=1}^N \phi_{ij} \left[\phi_{ji} \sigma_i^{C_r} + (1 - \phi_{ji}) \sigma_i^G \right] (1 - \text{step}(\xi_j - \xi_i)) \quad (5)$$

where $\beta_i^{C_t}$ and β_i^C represent the benefits of the agent i for each *cooperation* and *competition* interaction, respectively, and $\sigma_i^{C_r}$ and σ_i^G represent the costs of the agent i for each *cooperosity* and *generosity* interaction, respectively.

An interesting interpretation of (3) can be obtained by assuming $\beta_i^{C_t} = \beta_i^C = \bar{\beta}_i$ and $\sigma_i^{C_r} = \sigma_i^G = \bar{\sigma}_i$. In this case, if the agent i is a pure selfish, i.e., $d_{ij}^G = 0$ for all j , then from (4)–(5) and (3) it follows that $f_i = \bar{\beta}_i$ and does not depend on the number of connected agents. Analogously, for a pure altruist agent one can set $d_{ij}^C = 0$ for all j , then from (4)–(5) and (3) it follows that $f_i = -\bar{\sigma}_i$, i.e., the agent has only costs and no benefits.

3 PWL form of the opinion dynamics model

The model (1) can be written in a PWL form [8]. Indeed, for each combination of the influence functions values, (1) is a linear time invariant model which can be rewritten in the matrix form

$$\dot{\xi} = F_s \xi \quad (6)$$

with

$$F_s = \begin{pmatrix} -\sum_{j=1}^N \phi_{1j} & \phi_{12} & \dots & \phi_{1N} \\ \phi_{21} & -\sum_{j=1}^N \phi_{2j} & \dots & \phi_{2N} \\ \vdots & \vdots & \dots & \vdots \\ \phi_{N1} & \phi_{N2} & \dots & -\sum_{j=1}^N \phi_{Nj} \end{pmatrix} \quad (7)$$

for $s = 1, \dots, S$, and S is the total number of state space polyhedral regions corresponding to all the feasible combinations of the influence functions values. The number of functions ϕ_{ij} is $N_\phi = N(N - 1)$.

In the case of a static graph the influence functions ϕ_{ij} are fixed to 1 if the corresponding agents are connected and to 0 otherwise. As a consequence the matrix F_s is constant and it is the opposite of the classical Laplacian matrix.

Each index s corresponds to a polyhedral region of the state space and it can be represented by means of inequalities which depend on ϕ_{ij} . To this aim let us define the canonical vector e_i which has all entries equal to 0 except for the i -th element which is equal to 1. Then the expression (2) can be rewritten as

$$\phi_{ij}(\xi) = \begin{cases} 1, & \text{if } \begin{pmatrix} e_j^\top - e_i^\top \\ e_i^\top - e_j^\top \end{pmatrix} \xi \leq \begin{pmatrix} d_{ij}^C \\ d_{ij}^G \end{pmatrix} \\ 0, & \text{if } (e_i^\top - e_j^\top)\xi \leq -d_{ij}^C \\ 0, & \text{if } (e_j^\top - e_i^\top)\xi \leq -d_{ij}^G \end{cases} \quad (8)$$

for $i = 1, \dots, N, j = 1, \dots, N$. The formulation (8) induces a partition of the state space $[0, 1]^N$ into polyhedral regions, each one corresponding to a feasible combination of the influence functions ϕ_{ij} . In particular the polyhedra are defined by

$$D_s \xi \leq \delta_s \quad (9)$$

$s = 1, \dots, S$, where $D_s \in \mathbb{R}^{(N_s+2N) \times N}$ and $\delta_s \in \mathbb{R}^{(N_s+2N)}$ are constant matrices which can be obtained by collecting the $N_s \leq 2N_\phi$ independent inequalities deriving from (8) and the $2N$ inequalities corresponding to the state boundaries $0 \leq \xi_i \leq 1, i = 1, \dots, N$. The expression (9) is an \mathcal{H} -representation of the s -th polyhedron. A classical heterogeneous HK model assumes that the connectivity thresholds are not dependent on the direction of the connection [6], i.e.,

$$d_{ij}^C = d_{ji}^G, \quad d_{ij}^G = d_{ji}^C, \quad (10)$$

which in our vision can be interpreted as i being competitive (generous) versus j so as j is generous (competitive) versus i . Under the assumptions (10), the condition $\phi_{ij} = \phi_{ji}$ holds and the matrix F_s is symmetric.

Therefore, from (1) it follows that the sum of the states time derivatives is identically zero and the agents attitudes preserve their average for any time instant. We consider the general case where (10) do not hold. Since we are still interested in the convergence analysis to a consensus, it is useful to introduce a state transformation

which has the origin as an equilibrium point. Let us introduce the opinions differences

$$x_i = \xi_i - \xi_N, \tag{11}$$

for $i = 1, \dots, N - 1$. Any difference of two opinions can be written as a linear combination of the variables (11). Indeed:

$$\xi_j - \xi_i = (\xi_j - \xi_N) - (\xi_i - \xi_N) = x_j - x_i \tag{12a}$$

$$\xi_j - \xi_N = x_j \tag{12b}$$

$$\xi_N - \xi_i = -x_i, \tag{12c}$$

for any $i = 1, \dots, N - 1, j = 1, \dots, N - 1$. By combining (11) and (12) together with (1), one obtains

$$\begin{aligned} \dot{x}_i &= \dot{\xi}_i - \dot{\xi}_N = \sum_{j=1}^{N-1} \phi_{ij}(\xi_j - \xi_i) + \phi_{iN}(\xi_N - \xi_i) - \sum_{j=1}^{N-1} \phi_{Nj}(\xi_j - \xi_N) \\ &= - \left(\sum_{j=1}^N \phi_{ij} + \phi_{Ni} \right) x_i + \sum_{j=1, j \neq i}^{N-1} (\phi_{ij} - \phi_{Nj}) x_j, \end{aligned} \tag{13}$$

for $i = 1, \dots, N - 1$.

The expression (8) can be rewritten in terms of the opinions differences (11). In particular by using (12a) the influence functions can be expressed as

$$\phi_{ij}(x) = \begin{cases} 1, & \text{if } \begin{pmatrix} e_j^\top - e_i^\top \\ e_i^\top - e_j^\top \end{pmatrix} x \leq \begin{pmatrix} d_{ij}^C \\ d_{ij}^G \end{pmatrix} \\ 0, & \text{if } (e_i^\top - e_j^\top)x \leq -d_{ij}^C \\ 0, & \text{if } (e_j^\top - e_i^\top)x \leq -d_{ij}^G \end{cases} \tag{14}$$

for any $i = 1, \dots, N - 1, j = 1, \dots, N - 1$ where $x \in [-1, 1]^{N-1}$ is the state vector of the opinions differences (11) and the canonical vectors e_i have dimension $N - 1$. By looking at (13) we need to define also the influence functions ϕ_{Nj} and ϕ_{iN} in terms of the opinions differences. By using (12b) in (8) one obtains

$$\phi_{Nj}(x) = \begin{cases} 1, & \text{if } \begin{pmatrix} e_j^\top \\ -e_j^\top \end{pmatrix} x \leq \begin{pmatrix} d_{Nj}^C \\ d_{Nj}^G \end{pmatrix} \\ 0, & \text{if } -e_j^\top x \leq -d_{Nj}^C \\ 0, & \text{if } e_j^\top x \leq -d_{Nj}^G \end{cases}, \tag{15}$$

for any $j = 1, \dots, N - 1$ and by using (12c) in (8) one obtains

$$\phi_{iN}(x) = \begin{cases} 1, & \text{if } \begin{pmatrix} -e_i^\top \\ e_i^\top \end{pmatrix} x \leq \begin{pmatrix} d_{iN}^C \\ d_{iN}^G \end{pmatrix} \\ 0, & \text{if } e_i^\top x \leq -d_{iN}^C \\ 0, & \text{if } -e_i^\top x \leq -d_{iN}^G \end{cases}, \quad (16)$$

for any $i = 1, \dots, N - 1$.

The formulation (14)–(16) induces a partition of the state space $[-1, 1]^{N-1}$ into polyhedral regions, each one corresponding to a feasible combination of the influence functions ϕ_{ij} . In particular the polyhedra are defined by

$$C_s x \leq \gamma_s \quad (17)$$

$s = 1, \dots, S$, where $C_s \in \mathbb{R}^{(N_s+2N-2) \times (N-1)}$ and $\gamma_s \in \mathbb{R}^{(N_s+2N-2)}$ can be obtained by collecting the N_s independent inequalities (14)–(16), and the $2(N - 1)$ inequalities corresponding to the state boundaries $-1 \leq x_i \leq 1, i = 1, \dots, N - 1$. The expression (17) is an \mathcal{H} -representation of the s -th polyhedron.

By collecting all (13) together with (14)–(16), we obtain the HK model on relative opinions in the following PWL from

$$\dot{x} = A_s x, \quad x \in X_s, \quad s = 1, \dots, S \quad (18)$$

where

$$X_s = \{x \in \mathbb{R}^{N-1} \mid C_s x \leq \gamma_s\} \quad (19)$$

$$A_s = \begin{pmatrix} -\sum_{j=1}^N \phi_{1j} - \phi_{N1} & \phi_{12} - \phi_{N2} & \dots & \phi_{1N} - \phi_{N,N-1} \\ \phi_{21} - \phi_{N1} & -\sum_{j=1}^N \phi_{2j} - \phi_{N2} & \dots & \phi_{2N} - \phi_{N,N-1} \\ \vdots & \vdots & \dots & \vdots \\ \phi_{N-1,1} - \phi_{N1} & \phi_{N-1,2} - \phi_{N2} & \dots & -\sum_{j=1}^N \phi_{N-1,j} - \phi_{N,N-1} \end{pmatrix}. \quad (20)$$

By comparing (20) with (7) it follows that the matrix A_s can be obtained by taking the first $N - 1$ rows and $N - 1$ columns of F_s and by subtracting to each column of this matrix the corresponding element of the last row of F_s :

$$A_s = F_s(1 : N - 1, 1 : N - 1) - F_s(N, 1 : N - 1) \otimes 1_{N-1}, \quad (21)$$

where 1_{N-1} is the $N - 1$ column vector with all ones. The origin of (18) corresponds to the consensus.

4 A sufficient condition for the consensus

In this section we propose a sufficient condition for the asymptotic stability of the origin of the PWL model (18)–(19). To this aim let us recall some definitions.

Given λ points $\{v_\ell\}_{\ell=1}^\lambda$, $v_\ell \in \mathbb{R}^n$, $\lambda \in \mathbb{N}$, a *conical hull*, say $\text{cone}\{v_\ell\}_{\ell=1}^\lambda$, is the set of points $v \in \mathbb{R}^n$ such that $v = \sum_{\ell=1}^\lambda \theta_\ell v_\ell$, with $\theta_\ell \in \mathbb{R}_+$; a *convex hull*, say $\text{conv}\{v_\ell\}_{\ell=1}^\lambda$, is the conical hull with $\sum_{\ell=1}^\lambda \theta_\ell = 1$. Each polyhedron $X_s \subset [-1, 1]^{N-1}$ in (19) can be equivalently represented by means of its \mathcal{V} -representation

$$X_s = \text{conv}\{v_{s,\ell}\}_{\ell=1}^{\lambda_s} \quad (22)$$

with $s = 1, \dots, S$. The vertices $\{v_{s,\ell}\}_{\ell=1}^{\lambda_s}$ of the polyhedron X_s can be obtained from the \mathcal{H} -representation (19) by using numerical tools, e.g., the tool `cddmex` in Matlab [4]. The conical hull of a polyhedron X_s represented as in (22) is the cone $\mathcal{C}_{X_s} \subset \mathbb{R}^{N-1}$ defined as

$$\mathcal{C}_{X_s} = \text{cone}\{v_{s,\ell}\}_{\ell=1}^{\lambda_s}. \quad (23)$$

Another cone of interest for our analysis, corresponding to the polyhedron X_s , is the cone generated by the homogenization procedure [9]:

$$\hat{\mathcal{C}}_{X_s} = \text{cone} \left\{ \begin{pmatrix} v_{s,\ell} \\ 1 \end{pmatrix} \right\}_{\ell=1}^{\lambda_s}. \quad (24)$$

Let us define the ray matrix $R_s \in \mathbb{R}^{(N-1) \times \lambda_s}$ of the cone $\mathcal{C}_{X_s} \subset \mathbb{R}^{N-1}$ corresponding to X_s as follows

$$R_s = \begin{pmatrix} v_{s,1} & \cdots & v_{s,\lambda_s} \end{pmatrix}, \quad (25)$$

and the matrix $\hat{R}_s \in \mathbb{R}^{N \times \lambda_s}$ given by

$$\hat{R}_s = \begin{pmatrix} v_{s,1} & \cdots & v_{s,\lambda_s} \\ 1 & \cdots & 1 \end{pmatrix}. \quad (26)$$

With reference to the partition $\{X_s\}_{s=1}^S$, say Σ_0 the subset of indices s such that $0 \in X_s$ and Σ_1 its complement, i.e., $\Sigma_0 \cup \Sigma_1 = \{1, \dots, S\}$.

The asymptotic stability to the consensus is studied by using a Lyapunov approach. Let

$$V(x) = x^\top P_s x + 2q_s^\top x + r_s, \quad x \in X_s, s = 1, \dots, S \quad (27)$$

be the candidate PWQ function, where $\{P_s\}_{s=1}^S$ are symmetric matrices with $P_s \in \mathbb{R}^{(N-1) \times (N-1)}$, $\{q_s\}_{s=1}^S$ are vectors with $q_s \in \mathbb{R}^{N-1}$, $\{r_s\}_{s=1}^S$ are real scalars.

An important aspect for our stability analysis is the continuity of (27). Consider the matrices $\{\hat{P}_s\}_{s=1}^S$ with $\hat{P}_s \in \mathbb{R}^{N \times N}$ given by

$$\hat{P}_s = \begin{pmatrix} P_s & q_s \\ q_s^\top & r_s \end{pmatrix}. \quad (28)$$

Say X_h and X_k two elements of $\{X_s\}_{s=1}^S$ such that $X_h \cap X_k \neq \emptyset$ and $\Gamma_{hk} \in \mathbb{R}^{N \times m_{hk}}$, $m_{hk} < N$ the matrix of the common rays of the corresponding cones $\hat{\mathcal{C}}_{X_h}$ and $\hat{\mathcal{C}}_{X_k}$

obtained by applying the homogenization procedure. If the following conditions hold

$$\Gamma_{hk}^\top (\hat{P}_h - \hat{P}_k) \Gamma_{hk} = 0 \tag{29}$$

for all $h, k \in \{1, \dots, S\}$, such that $X_h \cap X_k \neq \emptyset$, then (27) is continuous on the common boundary between X_h and X_k , see [8].

We consider the stability problem for the origin of the model (18). It is assumed that, for any initial condition, (18) has at least one solution in the sense of Caratheodory, i.e., there exists an absolutely continuous function $x(t) : [0, \infty) \rightarrow \mathbb{R}^{N-1}$ which satisfies (18) almost everywhere. We assume that the system does not present sliding modes and Zeno behaviours.

By using the results in [8] it is easy to get a sufficient condition for the local asymptotic stability of the origin of (18). The condition is formulated as the feasibility of a set of constrained linear matrix inequalities. Any solution of this set directly provides the matrices of a PWQ Lyapunov function.

Theorem 4.1. *Consider the PWL system (18) with the polyhedra $\{X_s\}_{s=1}^S$ expressed as (22) and the PWQ function (27) as a candidate Lyapunov function. Consider the matrices $\{R_s\}_{s \in \Sigma_0}$ with $R_s \in \mathbb{R}^{(N-1) \times \lambda_s}$ given by (25) and the matrices $\{\hat{R}_s\}_{s \in \Sigma_1}$ with $\hat{R}_s \in \mathbb{R}^{N \times \lambda_s}$ given by (26). Define the matrices*

$$\hat{A}_s = \begin{pmatrix} A_s & 0_{N-1} \\ 0_{N-1}^\top & 0 \end{pmatrix} \tag{30}$$

with $s \in \Sigma_1$. Consider the set of LMIs

$$R_s^\top P_s R_s - N_s \succcurlyeq 0 \tag{31a}$$

$$-R_s^\top (A_s^\top P_s + P_s A_s) R_s - M_s \succcurlyeq 0 \tag{31b}$$

for all $s \in \Sigma_0$, and

$$\hat{R}_s^\top \hat{P}_s \hat{R}_s - N_s \succcurlyeq 0 \tag{32a}$$

$$-\hat{R}_s^\top (\hat{A}_s^\top \hat{P}_s + \hat{P}_s \hat{A}_s) \hat{R}_s - M_s \succcurlyeq 0 \tag{32b}$$

for all $s \in \Sigma_1$, and the set of inequalities

$$2q_s^\top R_s e_h \geq 0, \quad 2q_s^\top A_s R_s e_h \geq 0 \tag{33}$$

for $h = 1, \dots, \lambda_s$, $s \in \Sigma_0$. If there exist symmetric matrices $\{P_s\}_{s=1}^S$, $\{q_s\}_{s=1}^S$, $\{r_s\}_{s \in \Sigma_1}$, symmetric (entrywise) positive matrices $\{N_s\}_{s=1}^S$ and $\{M_s\}_{s=1}^S$, such that the set of linear matrix inequalities (31), (32) subject to the equality constraints (29) and to the inequality constraints (33) has a solution, then the origin is locally asymptotically stable for any initial condition in the partition $\cup_{s=1}^S X_s$, provided it is an invariant set.

5 Numerical results

In this section some numerical simulations are illustrated and the proposed PWQ Lyapunon function approach is applied for the stability analysis of (18)–(19).

In Fig. 2 two different time evolutions of (1) with $N = 100$ are shown. A homogeneous scenario, with the same connectivity thresholds for all agents, that lead to clustering is shown in Fig. 2(a). By introducing a random amount of *generosity*, the agents reach the consensus so as shown in Fig. 2(b).

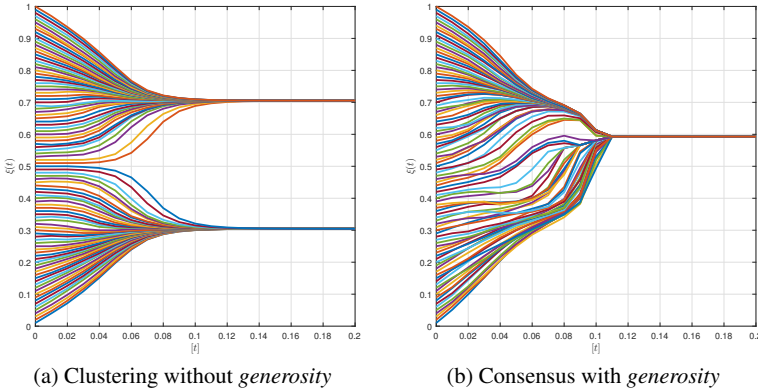


Fig. 2: Two different time evolutions of (1) with $N = 100$ and the same initial conditions (uniformly distributed in the interval $[0, 1]$): $d_{ij}^{C,G} = d_{ji}^{C,G} = 0.24$ (a); $d_{ij}^C = d_{ji}^C = 0.24$, $d_{ij}^G = d_{ji}^G$ randomly chosen with uniform distribution in the interval $[0.23, 0.25]$ (b).

Fig. 3 shows how *generosity* affects the average of the agents' opinions and the agents' fitness: by increasing the *generosity*, the corresponding averages increase, i.e., a larger benefit for all agents is obtained. This result confirms that more *generosity* leads to a larger benefit for the entire network [17], let us say an improved social capital. Viceversa, it can be shown that by increasing connectivity thresholds corresponding to the *competition*, the averages of the attitudes decrease.

Theorem 4.1 has been applied for the stability analysis of (18)–(19) with $N = 3$ and all thresholds equal to 0.5. By solving (29)–(33) with `Matlab` and `CVX` [5], a PWQ Lyapunon function has been obtained for the star-shape region contained in the feasibility domain shown in Fig. 4(a). By virtue of Proposition 2.1 in [14] and Proposition 3.3 in [24], the star-shape region in Fig. 4(a) is an invariant set, which allows to conclude the local asymptotic stability of the consensus. Fig. 4(b) shows a state trajectory and some level curves of the obtained PWQ Lyapunon function.

6 Conclusion

Starting from the heterogeneous HK model, a new opinion dynamics model has been proposed. The model is based on the concepts of *generosity* and *competition*,

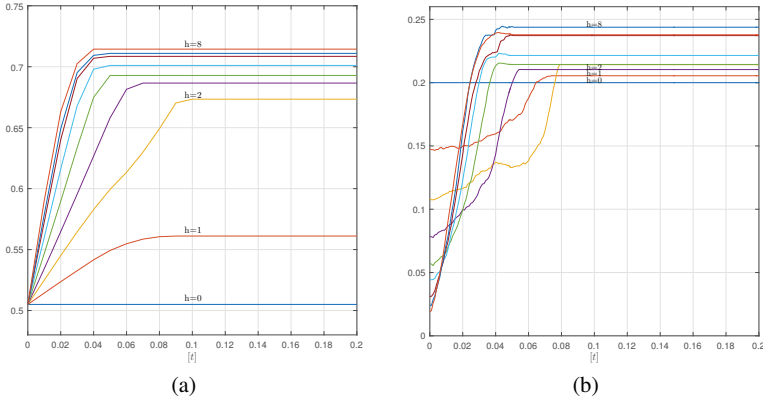


Fig. 3: Time evolutions for (1) with $N = 100$, initial conditions uniformly distributed in the interval $[0, 1]$, $d_{ij}^C = d_{ji}^C = 0.20$ and $d_{ij}^G = d_{ji}^G$ randomly chosen with uniform distribution in the intervals $[0.20, 0.20 + h/10]$ for $h = 0, \dots, 8$ identifying the 9 different simulations: opinions average (a); fitness average with $\beta_i^{C_t} = 1, \beta_i^C = 0.8, \sigma_i^{C_r} = 0.6, \sigma_i^G = 0.5, \forall i$ (b).

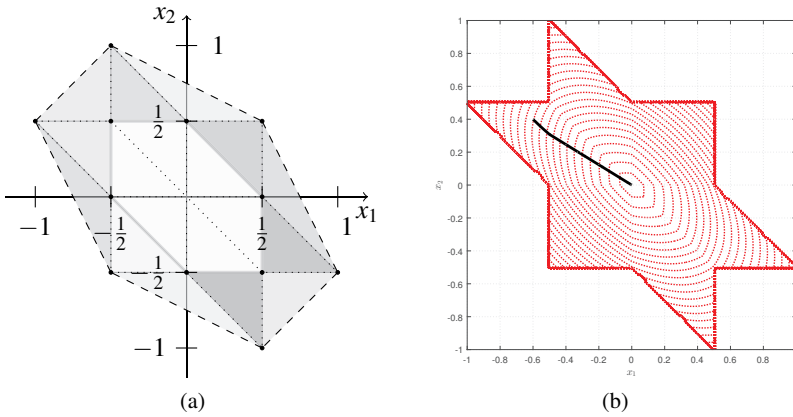


Fig. 4: State space for the model (18)–(19) with $N = 3$ and $d_{ij}^C = d_{ij}^G = 0.5, i = 1, 2, 3, j = 1, 2, 3$: polyhedral partition (19) (a); a state trajectory (black line) and some level curves of the PWQ Lyapunov function (dotted red lines) (b).

together with their combination with the cooperation between agents which leads to the *coopetition* and *cooperosity* behaviours. The model has been represented in a PWL form by using the state space polyhedra partition induced by the influence functions values. A PWQ Lyapunov function approach has been applied in order to determine a region of attraction of the consensus. Future work will investigate the validation of the proposed model through empirical data representing how attitudes dynamically evolve in human networks.

References

- [1] Blondel, V.D., Hendrickx, J.M., Tsitsiklis, J.N.: Continuous-time average-preserving opinion dynamics with opinion-dependent communications. *SIAM Journal on Control and Optimization* **48**(8), 5214–5240 (2010)
- [2] Ceragioli, F., Lindmark, G., Veibäck, C., Wahlström, N., Lindfors, M., Altafini, C.: A bounded confidence model that preserves the signs of the opinions. In: *Proc. of European Control Conference*, pp. 543–548. Aalborg, Denmark (2016)
- [3] Friedkin, N.E.: The problem of social control and coordination of complex systems in sociology: A look at the community cleavage problem. *Control Systems Magazine* **35**(3), 40–51 (2015)
- [4] Fukuda, K.: CDD. Swiss Federal Institute of Technology, https://www.inf.ethz.ch/personal/fukudak/cdd_home/
- [5] Grant, M., Boyd, S.: CVX: Matlab software for disciplined convex programming, version 2.1. <http://cvxr.com/cvx> (2014)
- [6] Hegselmann, R., Krause, U.: Opinion dynamics and bounded confidence models, analysis, and simulation. *Journal of Artificial Societies and Social Simulation* **5**(3), 1–33 (2002)
- [7] Hu, J., Zheng, W.X.: Emergent collective behaviours on coepetition networks. *Physics Letters A* **378**(26–27), 1787–1796 (2014)
- [8] Iervolino, R., Tangredi, D., Vasca, F.: Cone-copositivity for absolute stability of Lur’e systems. In: *Proc. of European Control Conference*, pp. 549–554. Aalborg, Denmark (2016)
- [9] Iervolino, R., Vasca, F.: Cone-copositivity for absolute stability of Lur’e systems. In: *Proc. of 53rd IEEE Conference on Decision and Control*, pp. 6305–6310. Los Angeles, California, USA (2014)
- [10] Iervolino, R., Vasca, F., Iannelli, L.: Cone-copositive piecewise quadratic Lyapunov functions for conewise linear systems. *IEEE Transactions on Automatic Control* **60**(11), 3077–3082 (2015)
- [11] Liang, H., Yang, Y., Wang, X.: Opinion dynamics in networks with heterogeneous confidence and influence. *Physica A* **392**(9), 2248–2256 (2013)
- [12] Lorenz, J.: Continuous Opinion Dynamics under bounded confidence: A Survey. *Int. Journal of Modern Physics C* **18**(12), 1819–1838 (2007)
- [13] Meng, Z., Shi, G., Johansson, K.H., Cao, M., Hong, Y.: Behaviors of networks with antagonistic interactions and switching topologies. *Automatica* **73**, 110 – 116 (2016)
- [14] Motsch, S., Tadmor, E.: Heterophilious dynamics enhances consensus. *SIAM Review* **56**(4), 577–621 (2014)
- [15] Newman, M.E.J.: *Networks. An Introduction*. Oxford University Press, Oxford, UK (2010)
- [16] Nowak, M.A.: Five rules for the evolution of cooperation. *Science* **314**(5805), 1560–1563 (2006)
- [17] Nowak, M.A., Coakley, S.: *Evolution, Games, and God: The Principle of Cooperation*. Harvard University Press, Cambridge, Massachusetts (2013)
- [18] Olfati-Saber, R., Murray, R.M.: Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control* **49**(9), 1520–1533 (2004)
- [19] Ren, W., Beard, R.W.: Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control* **50**(5), 655–661 (2005)
- [20] Scafuti, F., Aoki, T., di Bernardo, M.: Heterogeneity induces emergent functional networks for synchronization. *Physical Review E* **91**(6), 1–6 (2015)
- [21] Valcher, M.E., Misra, P.: On the consensus and bipartite consensus in high-order multi-agent dynamical systems with antagonistic interactions. *Systems and Control Letters* **66**, 94–103 (2014)
- [22] Vasca, F.: #cooperosity = cooperation + generosity: a dynamic receipt for being a system, a network, a community. *Twitter* **24 Aug, 14:06** (2016)

- [23] Yami, S., Castaldi, S., Dagnino, G.B., Roy, F.L.: *Coopetition. Winning strategies for 21st century*. Edward Elgar Publishing, Cheltenham, UK (2010)
- [24] Yang, Y., Dimarogonas, D., Hu, X.: Opinion consensus of modified Hegselmann-Krause models. *Automatica* **50**(2), 622–627 (2014)

Effect of Direct Reciprocity on Continuing Prosperity of Social Networking Services

Kengo Osaka, Fujio Toriumi and Toshiharu Sugawara

Abstract This paper investigates the effect of direct reciprocity on voluntary participation in social networking services (SNS) by modeling them as a type of public goods (PG) game. Because the fundamental structure of SNS is similar to the PG games, some studies have focused on why voluntary activities in SNS emerge by modifying the PG game. However, their models do not include direct reciprocity between users, even though it is known that reciprocity is a key mechanism to maintain and evolve cooperation in human society — one that is actually observed on SNS. To analyze the effect of reciprocity on SNS, we first developed an abstract model of SNS called reciprocal rewards and meta-rewards games that are extensions of the PG game. Then, we conducted experiments to understand how reciprocity facilitates cooperation by examining the proposed games using complete-graphs, WS networks, and a Facebook network. Finally, we analyze the findings derived from our experiments using the reciprocal rewards games and propose the concept of half free-riders to explain what maintains cooperation-dominant situations.

1 Introduction

Many people use one or a few social networking services (SNS) such as Twitter, Facebook, and Google+, not only to share and exchange local information among limited specialized and close-friend groups but also to publish/obtain public information for the purposes of opinion exchange, advertising, marketing, and political participation/campaigns [13]. SNS are usually run by companies and organizations but cannot persist without a huge amount of updated content posted continuously by individual users. However, the mechanism that leads to such continual posting

Kengo Osaka (e-mail: k.osaka@isl.cs.waseda.ac.jp)✉ · Toshiharu Sugawara (e-mail: sugawara@waseda.jp)✉
Waseda University, Tokyo 1698555, Japan,

Fujio Toriumi (e-mail: tori@sys.t.u-tokyo.ac.jp)
The University of Tokyo, Tokyo 1138656, Japan

activities is not well known, since such user activities incur various costs and effort in terms of creating and submitting the content. In addition, some *free riders* (or *lurkers*) exist, that is, users that just read the content and never post articles. To provide incentives to individual users to keep submitting content, many SNS have introduced a number of specific mechanisms, such as providing comments on articles, comments on comments, the number of followers, signs showing articles have been read, and “Like” buttons. These mechanisms can provide quantitative rewards (e.g., showing the numbers of readers and followers) as well as psychological rewards that provide feelings of connection to people and a sense of belonging [9]. However, these incentives also rely on users’ voluntary behavior and incur some cost and time on them.

As the variety of social media on the Internet continues to spread all over the world, it is an important issue to identify the conditions, mechanisms, and/or design methodology inherent to an active and thriving SNS. One approach to this end is based on an evolutionary game theoretic approach. For example, Toriumi et al. [15] and Hirahara et al. [6] discussed mechanisms to keep SNS active by using an evolutionary game on a variety of network structures. They proposed a *rewards game* (RG) and *meta-rewards game* (MRG), which were dual parts of Axelrod’s meta-norms game, and their own extension, called an SNS-norms game [6], to identify evolved behaviors of *agents* that are the model of SNS users. They then analyzed the conditions for a *cooperation dominant situation*, which corresponds to when SNS are active. They found that meta-rewards such as comments on article comments [15] and a simple (so, low-cost) response mechanism for rewards such as “Like” buttons for articles [7] play an important role in SNS. However, these studies did not consider social and personal relationships between peers. Furthermore, some SNSs have no mechanism to provide meta-rewards, and we believe that another mechanism also affects SNS activities.

Nowak [11] pointed out that one of five mechanisms — kin selection, direct and indirect reciprocity, network reciprocity, and group selection — is necessary for evolving cooperation in human society, and Rand and Nowak [12] showed the empirical evidence for human cooperation by these mechanisms. Such mechanisms also exist and play crucial roles in online networks [5, 8, 14]. For example, Faraj and Johnson [5] found that network exchange patterns in an online community are characterized by reciprocity patterns and are different from those characterized by preferential attachment [4]. Takano et al. [14] analyzed player action logs and found that cooperation based on reciprocity could be observed in a network game. We conclude that reciprocity, especially direct reciprocity, is essential in SNS because connections between users are usually established by direct interaction such as “comments on articles” and “comments on comments.”

Thus, we extended an existing abstract model of SNS [15] to examine the effect of direct reciprocity between users on continual and active use of SNS. The extended model is called a *reciprocity (meta-)rewards game* whose structure is similar to the (M)RG, but agents tag the peer agents and decide their behaviors on the basis of recent reciprocal behaviors of these peers. We then investigate why the rates of cooperation increase and when the established cooperation collapses. Our experimental results

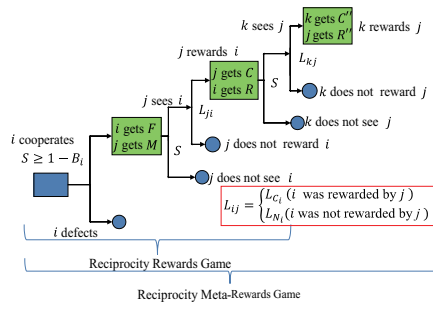
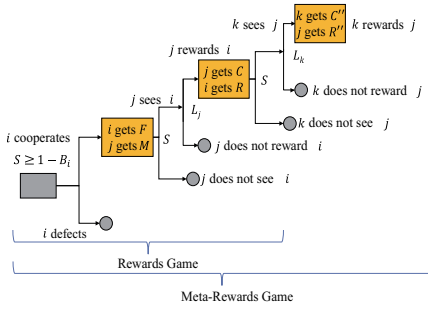


Fig. 1: Meta-rewards and rewards games. Fig. 2: Reciprocity (meta-)rewards game.

using complete and WS networks [16] suggest that users are cooperative not with all agents but with a few mutually close friends established on the basis of past reciprocal behavior. To explain this phenomenon, we propose the concept of *half free-riders* and discuss the interaction structure to maintain a cooperation-dominant situation that corresponds to a situation in which SNS continue to prosper. We also found that network structure affects the continuation of a cooperation-dominant situation.

2 Proposed Model for Social Networking Services

2.1 Reciprocity Reward and Meta-Rewards Games

SNS are sustainable only when many articles and comments on them are posted by and shared among anonymous participants. Although some cost in terms of personal time and effort is incurred, users can obtain some information by reading it and can receive responses that provide feelings of connectivity, empathy, and contentment. On the other hand, there are many free riders who only read content. Therefore, SNS have the properties of public goods that are produced and maintained by cooperation in the SNS community, and its game structure is essentially an n -person prisoner’s dilemma (PD) game. Toriumi et al. [15] proposed RG and MRG as dual games of norms and meta-norms games [3] (Fig. 1) and attempted to explain the mechanism of voluntary participation in SNS. Although they showed that meta-reward, which corresponds to “comments on a comment,” for example, is important in terms of providing incentives to continue voluntary participation, they ignored reciprocity, which is a crucial characteristic to understand the activities in SNS. Hence, we introduce the *reciprocity rewards game* (RRG) and *reciprocity meta-rewards game* (RMRG) by incorporating the reciprocal relationships among agents into the RG and MRG (see Fig. 2).

Let $A = \{1, \dots, n\}$ be the set of agents. Agents are connected with graph $G = (A, E)$, where E is the set of links between agents. The set of neighbor agents of $i \in A$ is denoted by $A_i \subset A$. Agents in an R(M)RG game select the strategy of either *cooperation* or *defect*. Cooperation indicates posting articles and comments, and defect indicates just reading them. A user who almost always selects defect is

called a *free rider*. Agent $i \in A$ has three learning parameters: the probability of cooperation (i.e., posting a new article) B_i , the probability of giving rewards (e.g., posting a comment on the article) to reciprocal agents L_{C_i} , and the probability of giving rewards to other (normal) agents L_{N_i} . We call B_i , L_{C_i} , and L_{N_i} the *posting article rate*, the *reciprocal comment rate*, and the *normal comment rate*, respectively. We also call both L_{C_i} and L_{N_i} the *comment rates* hereafter. To apply the genetic algorithm, we express each of these parameters as three bits, so it has a discrete value $0/7$, $1/7$, \dots , or $7/7$. This expression is identical to that used in the meta-norms game [3]. Agent i has the *memory for reciprocal agents* W_i , which is the set of neighbor agents that posted comments on i 's articles or i 's comments in the recent T_W rounds. The positive integer T_W is called the *memory length*.

An RRG or RMRG proceeds as follows. For $\forall i \in A$, parameter S_i^t ($0 \leq S_i^t \leq 1$) is defined randomly or with a certain method in the t -th round (t is a positive integer) when i is going to post an article. If $S_i^t \geq 1 - B_i$, i posts a new article with cost F and with probability S_i^t , and agent $\forall j \in A_i$ reads the article posted by i and gains reward M by reading it. Then, j proceeds to the next phase with probability S_j^t . Agent j comments on the article with probability L_{ji} , where $L_{ji} = L_{C_j}$ if $i \in W_j$; otherwise $L_{ji} = L_{N_j}$. Then, j pays cost C , and i gains reward R through j 's comment. The game chain so far is referred to as the RRG.

Subsequent to the RRG, $k \in A_i$ reads j 's comment and proceeds to the next phase with probability S_k^t . If this happens, k posts a response to the comment with probability L_{kj} , where $L_{kj} = L_{C_k}$ if $j \in W_k$ and $L_{kj} = L_{N_k}$ if $j \notin W_k$. When k posts it, k pays cost C'' and j gains reward R'' . Then, the RMRG ends here. All agents perform this game once in a round. Note that because rewards and meta-rewards games do not take into account reciprocity, agents have only L_{N_i} (which is denoted by L_i in RG) and $W_k = \emptyset$.

2.2 Evolution by Genetic Algorithm

RRG and RMRG are evolutionary games, as are (meta-)norms and (meta-)rewards games. We define one generation of the game as the term in which all agents have four chances to post articles. At the end of one generation, each agent selects two agents as parents from its neighbors on the basis of *fitness values*, which are defined as the cumulative rewards received minus the cumulative costs incurred during the current generation. This process is continued up to a certain generation.

Each of three learning parameters, B_i , L_{C_i} , and L_{N_i} , is represented in three bits gene. The initial values of the nine bits genes are set randomly (agents in RG have six bits genes). The evolution consists of three phases: (1) selection of parents, (2) crossover, and (3) mutation. A child agent of i for the next generation is then generated as follows. First, in the parent selection phase, i selects two parent agents from i and i 's neighbor agents on the basis of the probability distribution $\{\Pi_j | j \in A_i \cup \{i\}\}$ defined as

$$\Pi_j = (v_j - v_{min})^2 / \sum_{k \in A} (v_k - v_{min})^2, \quad (1)$$

Table 1: Parameter values used in experiments.

Parameter		Value	Parameter		Value
Cost of posting article	F	-3.0	Cost of comment	C	-2.0
Reward for reading article	M	1.0	Reward for receiving comment	R	9.0

where A_i is the set of the neighbor agents of i , v_k is the fitness value of agent $k \in A$, and $v_{min} = \min_{i \in A} v_i$. Then, two new genes are generated using uniform crossover from the genes in the selected parent agents and one of them is randomly selected in the crossover phase. In the mutation phase, each bit of the gene of the child agent is inverted with the probability of 0.005. This means that if there are 20 agents in the network, 0.9 bits will mutate on average. After that, the derived gene is used for the child agent of i .

3 Experiments and Discussion

3.1 Experimental Setting

Our experiments focus on the RRG, since rewards by comment on a comment seem small and thereby insignificant in SNS. Furthermore, some simple response mechanisms (rewarding mechanisms), such as “Like” buttons and “read” icons, have no mechanism to give meta-rewards for posting articles. We rather think that reciprocity is more significant in SNS. We compare the results of RRG with those of RG [15] to investigate the features of the reciprocity rewards game. We investigate how reciprocity affects user behavior on SNS in our experiments. For this purpose, we compare the transitions of the average rates of cooperation, that is, posting articles or comments, in RRG with those in RG on complete graphs that have 20 nodes (so its average degree is 20), WS networks [16] that have 1000 nodes (average degree is 20), and an instance of a Facebook network [1] that has 4039 nodes. Note that in RRG, agents separately manage comment rates for reciprocal and normal agents in their genes. The parameter values we set in these experiments are listed in Table 1. These parameter values are determined on the basis of the experiments of Axelrod [3] and Toriumi et al. [15] and to compare our results with theirs. Note that the experimental data below are the average values of 20 independent experimental runs based on the different random seeds.

3.2 Effect of Reciprocity on Cooperation

Figures 3 and 4 indicate how the probabilities of posting an article and a comment varied over generation in the complete graph. Note that the average posting article rate B is defined as $\sum_{i \in A} B_i / |A|$, the average reciprocal comment rate $L_C = \sum_{i \in A} L_{C_i} / |A|$,

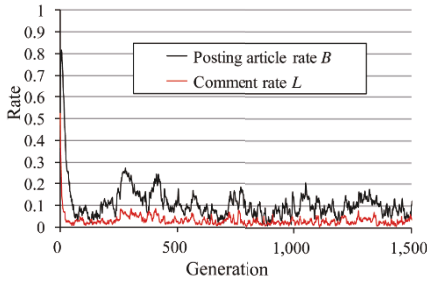


Fig. 3: Posting article and comment rates in RG (complete graph).

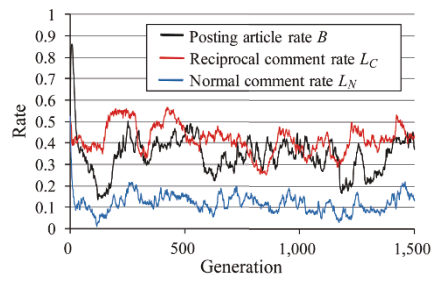


Fig. 4: Posting article and comment rates in RRG (complete graph).

and the average normal comment rate $L_N = \sum_{i \in A} L_{N_i} / |A|$. We show B and the average comment rate $L = \sum_{i \in A} L_i / |A|$ in RG.

In RG (Fig. 3), B and L transition at approximately 0.17 and 0.05, respectively. On the other hand, B and L_C transition at approximately 0.34 and 0.42, and L_N transitions at approximately 0.12 in RRG (Fig. 4). These results indicate that the values of B , L_C , and L_N in RRG were larger than those in RG. Thus, by taking into account reciprocity to decide the behavior, the activity in SNS improves in the complete graph, although that improvement is limited.

In the WS networks, we observe quite different phenomena. As shown in Figs. 5, the ratios of B , L_C , and L_N kept relatively higher values when $0 \leq p \leq 0.1$, where p is the *re-wiring probability* in the WS model. However, when $p = 0.3$ and 0.5 , the values of B became lower and fluctuated more. In particular, when $p = 0.5$ (Fig. 5(f)), B was close to that of the complete graph (Fig. 4). We also plotted in Fig. 6 how the average rates of B , L_C , and L_N changed in accordance with p to examine the effect of re-wiring probability on the agent's activity. Note that the WS model generates a regular graph when $p = 0$, whereas it generates random networks when $p = 1$ [10]. Their cluster coefficients are low when $p > 0.1$, so the small-world property with a high cluster coefficient only appears when $p \leq 0.1$. From these experimental data, we can say that in WS networks with small-world property and high-cluster coefficients, cooperation was dominant, but with the increase of re-wiring probability ($p > 0.1$), dominance of cooperation became weaker, and finally, B was around 0.3, which was slightly smaller than that of complete graphs.

Finally, we conducted the same experiments using an actual Facebook network [1]. This result, plotted in Fig. 7, indicates that posting article rate B was around 0.88, which is close to that of WS networks with p between 0.1 and 0.2.

3.3 Analysis of Phenomena

To understand more clearly why B , L_C , and L_N increased in RRG more than in RG (although the increases were sometimes limited), we investigated the results of one experimental trial of RG and RRG. Figures 8 and 9 show the results of the RG

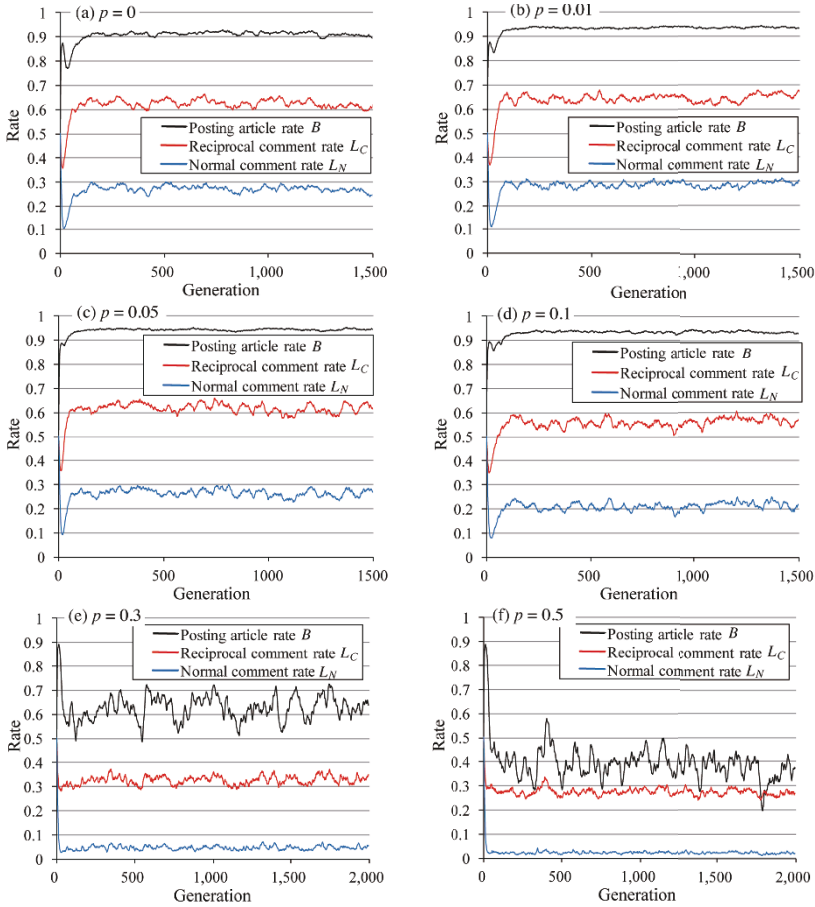


Fig. 5: Posting article and comment rates in RRG (WS networks).

and RRG in the complete network. Figure 8 indicates that the posting article and comment rates, B and L , rose temporarily and then immediately dropped in RG. Such temporary cooperation was caused by mutation. However, RG cannot maintain cooperation because it has no meta-reward mechanism and so has no incentive to comment on articles. This also caused agents to lose the incentive to post new articles, and cooperation therefore disappeared immediately. We also found that B , L_C , and L_N temporarily increased and then dropped in RRG. In both games, cooperation could not last for long, so their average values became small.

However, if we compare these figures more carefully, we can observe the difference between RG and RRG. Figure 8 indicates that B in RG occasionally increased to approximately 0.85 – 0.9 but did not reach 1.0. The value of L also increased but was much lower than that of B . We can explain this situation as follows. Some agents might have the genes to post articles comments by mutation, so their fitness values might slightly increase, and their genes spread to some degree. However, the RG has

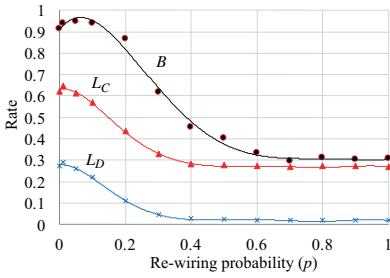


Fig. 6: Posting article rates in WS network with an approximate polynomial curve.

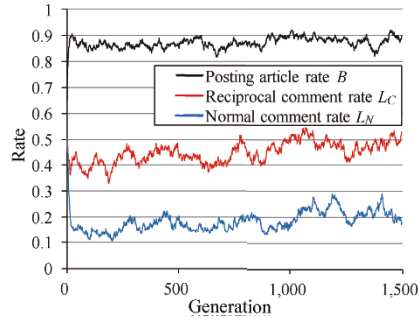


Fig. 7: Posting article rates (a Facebook network).

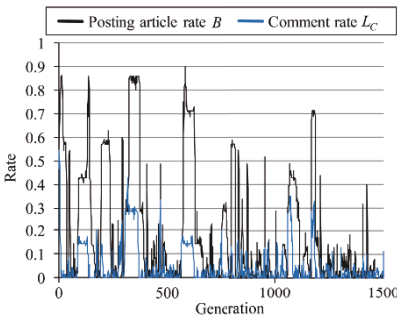


Fig. 8: Posting article and comment rates in RG (one trial).

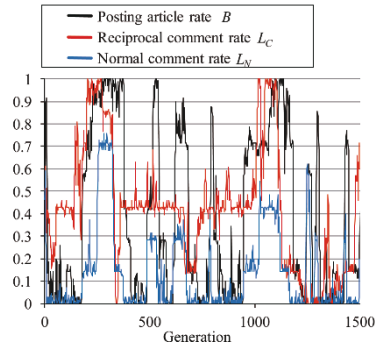


Fig. 9: Posting article and comment rates in RRG (one trial).

no incentive for giving comments (meta-rewards); agents with relatively large L also had low fitness values, so the value of L did not increase that much. After that, $B > L$ was held, and thereby the agents that post articles could not earn sufficient rewards, their fitness values decreased, and cooperation easily collapsed.

On the other hand, in RRG, B , L_C , and L_N rose intermittently for the same reason as the RG, but B and L_C reached 1.0 and lasted for a short period, as shown in Fig. 9; this means that almost all agents cooperate (posting articles) and give comments on cooperators' articles during this term. Furthermore, the value of L_C rarely dropped to zero. The difference between the RG and RRG is that, in the RRG, agents distinguish reciprocal agents from other agents and so can behave differently. Thus, agent i with high L_C comments selectively only on articles posted by reciprocal agents who commented on past articles posted by agent i . Such selective comments can prevent the collapse of cooperation by reducing the cumulative cost for comments. However, such prevention of collapse works *only when* $L_C > L_N$ and L_N is low; otherwise, many agents begin to comment on arbitrary articles without rewards and to have many reciprocal agents. This led to the game structure similar to RG, resulting in the high cost (no incentive to comment) and the collapse of cooperation.

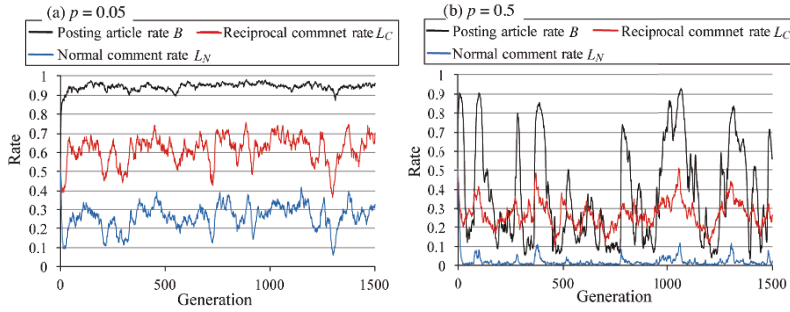


Fig. 10: Posting article and comment rates in RRG (one trial, WS networks).

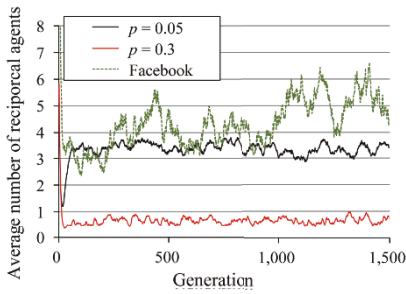


Fig. 11: Average number of reciprocal agents over generations (WS and Facebook networks).

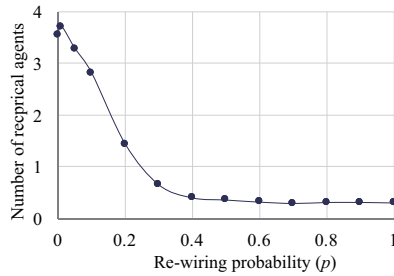


Fig. 12: Average number of reciprocal agents $0 \leq p \leq 1$.

In WS networks with small-world property and high-cluster coefficients, L_N maintained lower (around 0.25) and L_C maintained around 0.6, as shown in Fig. 10(a), which plots B , L_C , and L_N in WS networks with $p = 0.05$. This situation could balance between the costs and rewards, i.e., keep the balance in the numbers of posting articles and comments that incur some costs to contributors and rewards to receivers in the network. Such a condition could keep the posting article rate B higher. However, in WS networks with $p > 0.1$, B was considerably fluctuated, so the average became smaller, as shown in Fig. 10(b,) which shows the WS networks with $P = 0.5$ as an example. Furthermore, L_N was near 0, and L_C was also smaller than that in Fig. 10(a). The distribution of B seemed to be correlated with that of L_C and L_N , and we think that L_N affected agent’s activity more strongly.

To analyze this situation, we investigated the average number of reciprocal agents that each agent has in WS networks with $p = 0.05$ and 0.5, which is plotted in Fig. 11. Figure 12 also plots the average number of reciprocal agents in WS networks ($0 \leq p \leq 1$). Figure 11 indicates that when $p = 0.05$, agents had three to four reciprocal agents, and thus the reciprocity affected the agent’s strategies in RRG. However, when $p = 0.5$, agents had a small number of reciprocal agents, so their strategies are affected more by the normal agents. This structure of mutual effect is also identical to that of RG. Therefore, when $p = 0.5$, even if B temporary decreased,

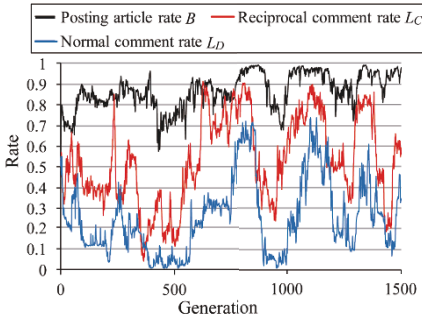


Fig. 13: Posting article and comment rates in RRG (one trial, a Facebook network).

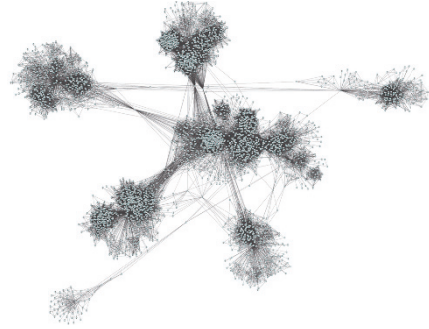


Fig. 14: Facebook network structure.

a small number of reciprocal agents continue to comment on articles ($L_C \approx 0.3$) and B decreased (but did not reach zero). When L_N became large by mutation, B increased again. However, because posting articles required some cost and agents could not receive sufficient comments, agents stopped posting them. Thus, we can say the reciprocity can maintain the dominance of cooperation in RRG if agents have an appropriate number (in our case, between 3 and 4) of reciprocal neighbors. However, unlike complete graphs, L_N was always less than half of L_C in WS networks (Fig. 6); the reason behind this phenomenon requires further analysis.

In a Facebook network (Fig. 13), the curves had the properties observed in Fig. 10(a) and (b), but are more complicated because its network consists of a number of communities that individually have their own sub-structures, as shown in Fig. 14, which is the visualized Facebook network used in this experiments. Though B , L_C , and L_N are fluctuated, the positing article rate B and the number of reciprocal agents maintained higher values (Figs. 11 and 13). Thus, we can say that separation of reciprocal and normal agents contributed to the thriving of RRG in this network.

3.4 Discussion

We explain what the phenomena described above in the RRG correspond to in actual SNS. When SNS users did not consider direct reciprocity when using SNS (that is, RG), users who often comment must stop commenting because RG has no incentive for comments. In RRG, when L_N is large, users take a similar strategy, i.e., users comments on many articles but have to pay high cumulative costs, and other agents stop posting articles. On the other hand, if individual users consider direct reciprocity to comment, they would comment on the articles of reciprocal users preferentially by looking at the content of memory, W_i . Furthermore, the number of reciprocal agents was not so large, only a few agents. Thus, when L_N is small, such selective comment behavior for receiving comments in future facilitates and maintains the norm for

cooperation. We also believe that condition $L_N \ll 1$ is a reasonable assumption in actual SNS.

If we look at our results from the macro-viewpoint, we have a number of suggestions related to the activity structure of SNS. First, agents seem to behave like *half free-riders* in cooperation-dominant situations. Agents gain rewards by reading articles of normal agents and do not pay for commenting on these articles. This part corresponds to free-rider's behavior. However, because only this behavior makes SNS inactive, agents heartily comment only on the articles posted by reciprocal agents. Agents have to pay some cost for these comments, but because the number of reciprocal agents is limited, we can keep the total cost lower. Of course, if agents post more comments on the articles posted by normal agents, the total cost increases and cooperation collapses.

Because we can assume that reciprocal agents are like close friends in human societies, the situation mentioned above is often exposed in actual SNS. A user, u , may have many peers, so u reads many articles posted by them. However, to gain the incentive to post articles, u has to receive not many but rather secure comments from u 's close peers. The articles posted by u are also read by many other users who can gain some rewards by behaving as free-riders for u . This relationship suggests the hierarchical (ego) structure observed in SNS [2], although the structure in our experiments is simpler. We will have to analyze the topological structure of close-friend relationships in RRG and compare it with the ego network observed in SNS.

Another interesting and remarkable phenomenon is that network structure affects the strategy for having close friends (so, securely interactive) or not in the RRG. In a certain type of network, like complete graphs and WS networks with high p values, posting article rate B was fluctuated and its average value became low. In this type of network, the number of reciprocal agents was low and their activity is mainly affected by only L_N . In contrast, in WS networks with low p ($0 \leq p \leq 0.1$), a high posting article rate was the dominant strategy and the cooperation was supported by high L_C . The Facebook network used in this experiment seems to be a collection of sub-networks that belong to the latter type of networks, where agents have appropriate numbers of close friends and posting articles/comments to each other is the dominant strategy. However, what characteristic of the network decides the agent strategy is still unknown, and this remains our future work.

4 Conclusion

We investigated the effect of reciprocity between users on the prosperity of SNS. For this purpose, we first proposed the reciprocal rewards game, which is an abstract model of SNS and an extension of the rewards game [15]. The structure of reciprocity between users is not included in the original rewards and meta-rewards games, although we believe that reciprocity affects the user's SNS activity. We conducted our experiments using complete graphs, WS networks, and a Facebook network to understand and analyze the effect of reciprocity on SNS activities on the evolution of cooperation. Our experimental results suggested that when the user behaved as a half

free-rider, meaning that the user behaved as a cooperator to a small number of reciprocal peers (close friends) but behaved as a free-rider to other peers (acquaintances), cooperation can evolve and be maintained.

We plan to investigate the characteristics of networks where it is more advantageous to have reciprocal agents to maintain secure interaction. In addition, we believe that interaction takes place not only between two users but also in a group of users, so we will include indirect reciprocity in our model in future.

Acknowledgement: This work is partly supported by JSPS KAKENHI (25280087).

References

- [1] Stanford Large Network Dataset Collection. <http://snap.stanford.edu/data/>
- [2] Arnaboldi, V., Conti, M., Passarella, A., Pezzoni, F.: Analysis of Ego Network Structure in Online Social Networks. In: ASE-IEEE Int. Conf. on Social Computing, pp31–40 (2012)
- [3] Axelrod, R.: An evolutionary approach to norms. *American political science review* **80**(04), 1095–1111 (1986)
- [4] Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. *Science* **286**, 509–512 (1999)
- [5] Faraj, S., Johnson, S.L.: Network exchange patterns in online communities. *Organization Science* **22**(6), 1464–1480 (2011).
- [6] Hirahara, Y., Toriumi, F., Sugawara, T.: Evolution of Cooperation in SNS-norms Game on Complex Networks and Real Social Networks. In: Social Informatics (SocInfo 2014), LNCS 8851, pp. 112–120. Springer (2014)
- [7] Hirahara, Y., Toriumi, F., Sugawara, T.: Cooperation-dominant Situations in SNS-norms Game on Complex and Facebook Networks. *New Generation Computing* **34**(3), 273–290 (2016).
- [8] Leider, S., Möbius, M.M., Rosenblat, T., Do, Q.A.: Directed altruism and enforced reciprocity in social networks. *The Quarterly Journal of Economics* **124**(4), 1815–1851 (2009).
- [9] Lin, H., Fan, W., Chau, P.: Determinants of users' continuance of social networking sites: A self-regulation perspective. *Information and Management* **51**(5), 595–603 (2014).
- [10] Newman, M.E.J.: The structure and function of complex networks. *SIAM Review* **45**, 167–256 (2003)
- [11] Nowak, M.A.: Five rules for the evolution of cooperation. *Science* **314**(5805), 1560–1563 (2006)
- [12] Rand, D.G., Nowak, M.A.: Human cooperation. *Trends in Cognitive Sciences* **17**(8), 413 – 425 (2013).
- [13] Stieglitz, S., Dang-Xuan, L.: Social media and political communication: a social media analytics framework. *Social Network Analysis and Mining* **3**(4), 1277–1291 (2013).
- [14] Takano, M., Wada, K., Fukuda, I.: Reciprocal altruism-based cooperation in a social network game. *CoRR abs/1510.06197* (2015). URL <http://arxiv.org/abs/1510.06197>
- [15] Toriumi, F., Yamamoto, H., Okada, I.: Why do people use social media? Agent-based simulation and population dynamics analysis of the evolution of cooperation in social media. *IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology*, vol. 2, pp. 43–50 (2012).
- [16] Watts, D., Strogatz, S.: Collective dynamics of 'small-world' networks. *Nature* **393**, 440–442 (1998)

Co-evolution of two networks representing different social relations in NetSense

Ashwin Bahulkar and Boleslaw K. Szymanski and Kevin Chan and Omar Lizardo

Abstract We examine the dynamics of co-evolution of two coupled social networks. The first is a cognitive network defined by nominations based on perceived prominence collected from repeated surveys of students during their first four semesters of college while the second is built from the behavioral network representing actual interactions between these individuals based on records of their mobile calls and text messages. We address three interrelated questions. First, we ask whether the formation or dissolution of a link in one of the networks precedes or succeeds formation or dissolution of the corresponding link in the other network (temporal dependencies). Second, we explore the causes of observed temporal dependencies between the two networks. For those temporal dependencies that are confirmed, we measure the predictive capacity of such dependencies. Finally, we examine whether there are systematic differences in the dissolution rates of symmetric (undirected) versus asymmetric (directed) edges in both networks. We find strong patterns of reciprocal temporal dependencies between the two networks. In particular, the creation of an edge in the *behavioral* network generally precedes the formation of a corresponding edge in the *cognitive* network. Conversely, the decay of a link in the cognitive network generally precedes a decline in the intensity of communication in the behavioral network. Finally, asymmetric edges in the cognitive network have lower overall communication volume and more asymmetric communication flows in the behavioral network.

Ashwin Bahulkar (e-mail: bahul@rpi.edu)✉ · Boleslaw K. Szymanski (e-mail: szymab@RPI.edu)✉

Rensselaer Polytechnic Institute, 110 8th St, Troy, NY 12180 Boleslaw K. Szymanski
Wroclaw University of Science and Technology, 50-370 Wroclaw, Poland

Kevin Chan (e-mail: kevin.s.chan.civ@mail.mil)
US Army Research Laboratory, Adelphi, MD 20783

Omar Lizardo (e-mail: olizardo@nd.edu)
University of Notre Dame, Notre Dame, IN 46556

1 Introduction

In this paper we investigate how two different social networks, one a *cognitive* network composed of subjective nominations and another a *behavioral* network composed of objectively recorded communications, relate to one another. We aim to understand in detail the relationship between these two networks, as the link between cognition and behavior is a long-standing, but understudied, problem in social network analysis [6, 7]. A key question in this literature is whether behavior precedes cognition, such that contacts with which we frequently interact become more cognitively salient, or whether cognition precedes behavior, such that we increase the amount of interaction with those contacts that we consider subjectively salient [2].

To make headway on these questions, we use a data source that contains dynamic information on both the cognitive salience of contacts and actual behavioral traces of communication behavior between individuals. We examine whether two social networks built from these different kinds of connections are temporally coupled. Our main hypothesis is that there exist reciprocal linkages between cognitive salience and behavioral communication with increasing communication leading to greater cognitive salience and with declining cognitive salience leading to the dissolution of behavioral edges [3].

To evaluate this hypothesis, we investigate whether increases in communication lead to increases in cognitive salience and whether cognitive salience is associated with increased communication behavior. We also examine whether declining cognitive salience leads to a gradual decrease in actual communication. Finally, we ask whether non-reciprocity in cognitive salience is associated with non-reciprocity in actual communicative interaction [6], and whether persons who are exposed to sustained asymmetries in communication are motivated to cycle through more persons in their cognitive salience network in search of reciprocal interactions [8].

2 NetSense Data and the Networks

In this section, we introduce the NetSense data [10] and the networks derived from it. The data was collected at the University of Notre-Dame. At the start the Fall semester in 2011, 200 of the incoming freshmen were enrolled in the NetSense study. Over 150 participated until their graduation in the Spring of 2015. Students participating in the study received free smartphones with unlimited voice and text plans as an incentive for participation. We obtained time-stamped logs of communication records for all study participants. These data contain information on the the date, time and duration (for calls) and character length (for text messages). Data for the first four semesters (lasting from the Fall of 2011 to the Spring of 2013) of the project was available for this study.

Students participating in the NetSense study list up to twenty contacts at the beginning of each semester. Students were asked to list the names of those people with whom they thought they the most time communicating or interacting with. Below, we refer to these contacts as friends. These friends could be inside or outside the NetSense

study. Because students were asked to also provide the primary phone number of each friend we can link each friend mentioned in the survey to the time-stamped smartphone data. Accordingly, We propose a model for analyzing co-evolution of multiple networks representing different kinds of social relations between nodes. The *behavioral* network consists of the behavioral edges based on communication records of both telephone calls and text messages between individuals. Weights on the edges in the behavioral network change everyday, depending on the volume of communication. The *cognitive* network includes cognitive edges that are based on (possibly asymmetric) nominations collected through the surveys. Edges in the cognitive network appear and disappear once per semester.

3 Related Work

A model to generate two social networks synthetically, with both the networks co-evolving, capturing the properties of both networks is introduced in [12]. A rapidly evolving network based on games is studied in [9]. Nodes in this network have varying incentives to build links. We observe similar behavior in the NetSense data, where certain edges have incentive to develop into an edge in one of the networks, while others do not. The co-evolution of edges in relation to individual behavior in school dormitories is investigated in [5]. The co-evolution of employee networks in organizations in relation to individual attitudes is studied in [7]. In contrast to these studies, we explore how two social networks co-evolve in time.

4 Analysis of Co-Evolution of NetSense Networks

We conduct several experiments on the NetSense data to study how the two networks co-evolve. We divide these experiments into two broad categories: analyzing precedence of dissolution and formation of edges in both the networks and analysis of asymmetric edges in each of the networks.

First, we deal with the question of whether the formation and dissolution of edges each of the networks studied (cognitive and behavioral) are systematically related to each other. To do so, we examine whether forming or increasing the strength of an edge in one network (e.g. behavioral) precedes a corresponding edge creation in the other (e.g. cognitive) network. We also study whether edge dissolution in one network is informative of a corresponding dissolution event in the other. For example, we can ask how often the emergence or strengthening of behavioral edges leads to the formation of cognitive edges in a subsequent semester. We look at factors that may cause edges to form or dissolve and then infer if there are any causal relationship between the two networks. For example, we observe that high levels of communication between edges in behavioral network is often associated with the formation of future cognitive edges. So we can infer that high communication volume in the behavioral network often leads to the appearance of subjectively meaningful ties in the cognitive network.

4.1 Does higher communication in behavioral network predict the appearance of edges in the cognitive network?

We start by investigating whether we can observe increases in communication between two people *before* an edge between them appears in the cognitive network. To this end, we measure the communication between students in the semester before one of them nominates the other as a friend in the survey, and ascertain whether there is a difference in previous communication volume between nodes that are subsequently connected in the cognitive network and those which are not. Table 1 lists these results. Figures 1a and 1b illustrate how number of calls and messages are distributed among to-be-formed and not-to-be-formed edges in the cognitive network.

Table 1: Difference in communication volume between nodes to-be-nominated and not-to-be-nominated as friends, and future friendship nominations based on volume of communication between the corresponding nodes.

Semester No.	to-be-nominated		not-to-be-nominated		Calls		Messages	
	No. Calls	No. Messages	No. Calls	No. Messages	Precision	Recall	Precision	Recall
Semester 1	40	407	5	58	70	82	78	88
Semester 2	52	782	6.5	105	72	74	72	70
Semester 3	18	248	4	41	73	75	78	80

We find that, indeed, edge weight in the behavioral network is a good predictor of whether an edge subsequently appears in the cognitive network. In the first semester, edges in which one of the participants subsequently nominates the other as a significant contact differ by a factor of 8 (in terms of calls) and by a factor of about 7 (in terms of text messages) from those in which no edge emerges. Similar differences can be observed for semesters 2 and 3.

We further examine whether edge weight in the behavioral network can be used to predict the appearance of future links in the cognitive network. Table 1 lists the results of these analyses. We find that we are able to predict a significant proportion of edges in the cognitive network using information from the behavioral network, about 70-80 %, with a reasonable recall [1]. The threshold that gives us the best balance between precision and recall can be found plotting ROC curves [2]. We infer that nomination as a friend is often preceded by high levels of communication between the corresponding nodes. Hence, there is strong reason to conclude that the dependence of the cognitive network on the behavioral network is causal.

4.2 Do edges in the cognitive network have stronger links in the behavioral network?

Next, we investigate whether we can observe significant differences in communication volume between two people once an edge appears in the cognitive network. To do

so, we compare the communication volume (the weight of the edge in the behavioral network) between nodes connected by the edges that appear in the cognitive network and those which do not.

Table 2: Difference between connected and disconnected edges in the cognitive network in terms of weight in the behavioral network and prediction of future friendship nominations based in the cognitive network on the volume of communication in the behavioral network.

Semester No.	Friends		Non-friends		Calls		Messages	
	No. Calls	No. Messages	No. Calls	No. Messages	Precision	Recall	Precision	Recall
Semester 1	70	667	7	72	71	76	61	84
Semester 2	41	915	12	190	70	70	61	78
Semester 3	74	1063	5	51	66	74	64	90
Semester 4	34	729	4	37	68	72	62	86

Table 2 shows the results. We observe a large difference in communication volumes between these two edge classes, with edges in which one person nominates the other as a friend displaying high levels of behavioral interaction. For instance, in the first semester, nodes connected by edges that were connected in the cognitive network differed from those that were not by a factor of 7 (for calls) and a factor of about 9 (for texts), with differences of similar magnitude holding for subsequent semesters.

We verify whether the volume of communication in the behavioral network can allow us to predict forming of an edge in the cognitive network. Table 2 shows that we can indeed predict a significant number of friendship nominations purely from communication volume in the behavioral network, about 70-90 %, with reasonable precision.

4.3 Do newly formed edges in the cognitive network differ from older edges in terms of communication levels between their nodes?

Next, we study how nodes connected by the newly formed and older links in the cognitive network differ in terms of their edge weight in the behavioral network. To this end, we measure the amount of communication between nodes joined by older (more than one semester) and newly formed (one semester) cognitive edges. We observe that cognitive edges joining nodes with higher communication levels nodes connected by than newer links in the friendship network. Table 3 lists these differences. Figures 2a and 2b illustrate how number of calls and messages are

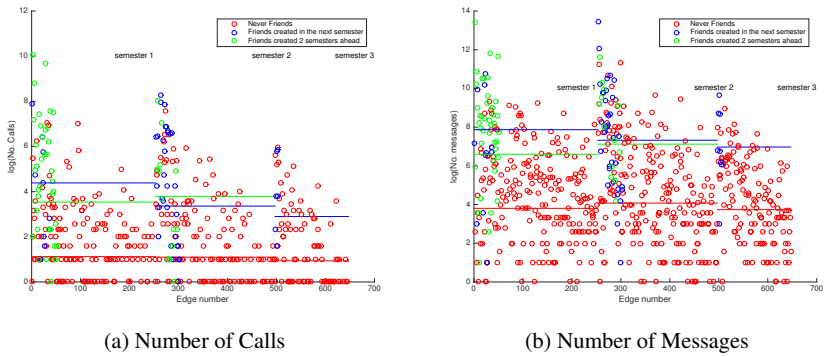


Fig. 1: Call and message volumes between to-be-friends in one semester (blue circles), to-be-friends in two semesters (green circles) and not-to-be-friends (red circles). Generally, to-be-friends have higher call and message volume than not-to-be-friends. The continuous lines show the average value for the circles of each color. The separation is large between red and green lines, red and blue lines, but small between blue and green lines. Most of the to-be-friends edges appear in the first and second semester, since very few new friendships are formed in the fourth semester.

distributed among pairs of nodes connected by to-be-formed and not-to-be-formed edges in the cognitive network.

Table 3: Difference behavioral communication volume between old and new edges in the cognitive network.

Semester No.	Newly observed nominations		Nominations older than one semester	
	No. Calls	No. Messages	No. Calls	No. Messages
Semester 2	6	57	61	1340
Semester 3	63	1026	172	2447
Semester 4	7	256	53	1067

We also observe that as these newly formed cognitive edges age, the nodes connected by them come to have communication volumes similar to, or perhaps slightly higher, than cognitive edges that have existed for a longer time. To shed further light on this issue, we examine communication volumes of cognitive edges in the 3rd and the 4th semesters, and we divide them into edges which were created in the 2nd and the 3rd semesters respectively, and edges which existed since the 1st semester. We call the former moderately old edges and the latter very old edges. We observe that moderately old edges carry on an average of 49 calls and 903 calls, while

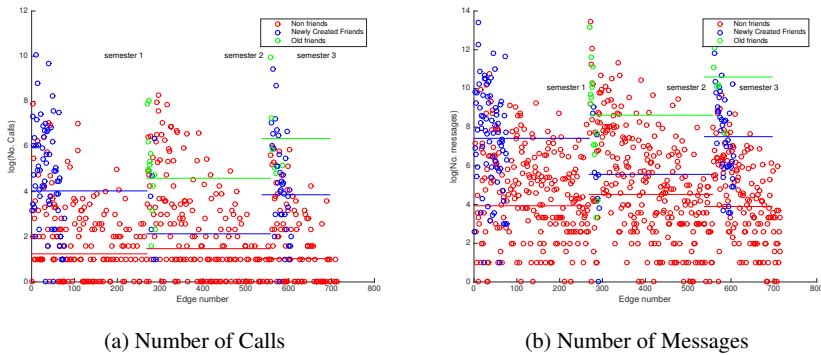


Fig. 2: Communication volumes nodes connected by old edges in the cognitive network (green circles), newly created edges in the same network (blue circles), and disconnected nodes (red circles). The continuous lines show the average value for the circles of the corresponding color in each semester. The separation is significant between all three lines. Generally, nodes connected by cognitive edges in which one person nominates the other as a friend have a higher communication volume. A significant number of persons that do not nominate each other, however, have high message volumes as well, but less so with the call volume.

very old edges exchange 29 calls and 795 messages. We infer that communication between nodes that are also connected in the cognitive network increases gradually, but then finally stabilizes over a period of time.

4.4 How likely does communication dissolve after the corresponding edge disappears in the cognitive network?

The next question we study is how likely are the communication links to dissolve after their corresponding cognitive edges dissolve. To assess that, we measure the rate at which dyads that dissolve their cognitive edges also dissolve the corresponding edges in the behavioral network, and compare that with the rate at which behavioral network links dissolve at random. We find that behavioral network dyads that first experience a dissolution event in the cognitive network are more likely to dissolve their behavioral edge than a random dyad does.

Let $BDCN$ denote the average link dissolution rate in the behavioral network for persons who are not connected in the cognitive network, and $BDCY$ denote the average link dissolution rate in the behavioral network for dyads that are connected in the cognitive network. In the third and fourth semesters, $BDCN$ is significantly greater than $BDCY$, while the reverse is observed in the second. We observe values

of 64%, 55% and 50% for BDCN for the three semesters, and 42%, 74% and 62% for BDCY. We also measure the rate at which the nodes connected by the cognitive edges that persist into the following semester dissolve their behavioral edges, and denote it as BDCP. We find that BDCP is always 0, meaning that if there is link persistence in the cognitive network then there is always link persistence in the behavioral network.

4.5 Patterns of communication decay following link dissolution in the cognitive network

Finally, we examine whether edge weights in the behavioral network decrease after links in the cognitive network dissolve. We measure this effect using the “recency” score [4], where recent communication has higher weight than older communication. If there is a decrease in communication, the recency weighted score will be lower than communication score without weights.

- **Recency Score (RS):** Each semester lasts 5 months; odd numbered lasts from August to December, while even numbered lasts from January to May. We assign weights to communication during each month in the following manner: -0.3 for the 1st month, -0.1 for the 2nd month, 0.1 for the 3rd month, 0.3 for the 4th month and 0.5 for the 5th month.
- **Non-Recency Score (NRS):** We assign equal weights of 0.1 to communication in any of the months. We compare how much nodes connected by dissolving and persistent edges differ on both of these scores.

In Table 4, we list RS and NRS scores (computed from the behavioral network) for nodes connected by dissolving and persistent edges in the cognitive network. We then take the average numbers of calls and messages for these categories and compute the ratio of numbers of calls/messages between nodes connected by persistent cognitive edges to numbers of calls/messages between nodes joined by dissolving cognitive edges. We observe that the ratio increases when RS is used. This means, there is a bigger difference when RS is used, which indicates that nodes connected by dissolving edges in the cognitive network have more communication in the behavioral network during earlier months than in the later months. However, we do not observe this trend in the first semester, since the friendships are still developing, and communication is most likely to be increasing for all friendships.

We could draw the inference that students who dissolve cognitive links are much more likely not to communicate with each other at all, leading to a complete dissolution of the communication edge.

Table 4: Difference between to-be-friends and non-to-be-friends.

Semester 1							
NRS	Dissolved	Persistent	Ratio	RS	Dissolved	Persistent	Ratio
No. Calls	5	13	2.6	No. Calls	6	10	1.8
No. Texts	51	137	2.7	No. Texts	90	121	1.4
Semester 2							
NRS	Dissolved	Persistent	Ratio	RS	Dissolved	Persistent	Ratio
No. Calls	1	10	10.0	No. Calls	0.4	8.3	20.8
No. Texts	18	109	6.1	No. Texts	3.1	195	62.9
Semester 3							
NRS	Dissolved	Persistent	Ratio	RS	Dissolved	Persistent	Ratio
No. Calls	7	8	1.1	No. Calls	2.2	5.3	2.4
No. Texts	56	151	2.7	No. Texts	47	185	3.9

4.6 Analysis of asymmetric friendship cognitive edges

The NetSense data consists of periodic surveys where students nominate up to twenty friends at the beginning of every semester. We examine cognitive edges that are asymmetric, where only one of the respondents marked the other as a friend. We observe whether the nodes connected by these asymmetric edges in the cognitive network exhibit different patterns of communication and survival probabilities of the edges in the behavioral network. We find that asymmetric cognitive edges differ significantly from symmetric edges in both of these respects. The following sections illustrate the differences between asymmetric and symmetric cognitive edges.

4.6.1 Do nodes joined by asymmetric behavioral edges become dissolve their behavioral edges faster?

First, we examine if nodes connected by asymmetric edges in the cognitive network are more likely to dissolve their edges in the behavioral network than nodes connected by symmetric (mutual) cognitive edges. We measure the survival probabilities of behavioral edges between nodes connected by asymmetric and symmetric edges across all semesters. We observed that nodes connected by asymmetric cognitive edges are significantly more likely to dissolve their communication edges than symmetric cognitive edges.

The dissolution probabilities of of communication edges between nodes connected by the asymmetric cognitive edges are higher than communication edges between nodes with mutual cognitive edges in all three semesters. We observe that nodes joined by asymmetric edges have a dissolution probability of their communication edges of 90%, 87.5% and 50% in each of the three semesters, while such probabilities for symmetric edges have a dissolution probability of are 72%, 66% and 16% in

each of the three semesters. We also observe an overall downward trend in the dissolution probability. Initially, these are very high for the first semester, but they decline steadily over time. However, even in the third semester nodes connected by asymmetric edges in the cognitive network are more than three times more likely to dissolve their communication edges in the behavioral network than nodes joined by symmetric cognitive edges.

4.6.2 Differences in Communication volumes between asymmetric and symmetric edges

Next, we examine if nodes connected by asymmetric and symmetric cognitive edges differ in communication volume in the behavioral network. As shown in Table 5, we observe that, apart from the first semester, there is a significant difference between asymmetric and symmetric edges, with symmetric edges communicating more. In the first semester, the same difference exist, but it is much smaller and visible only if the sum of calls and messages is taken into account.

4.6.3 Asymmetric cognitive edges and asymmetric communication edges

Next, we examine whether nodes connected by asymmetric edges in the cognitive network are also more likely to have asymmetric communication patterns in behavioral network as well. We define “asymmetric” edge in the the behavioral network whenever we observe one node initiating communications with the other node more often than the reverse. We compare communication imbalance between nodes connected by asymmetric cognitive edges and nodes joined by symmetric edges of this type. We find that symmetric edges always have less asymmetrical communication patterns in the behavioral network than asymmetric cognitive edges. To measure asymmetry in communication, we first compute the ratio of the volume of communication in which the source node is the initiator to the volume of communication in which the destination node is the initiator: we call this quantity *OSC*. We multiply the number of calls by 10, since messages are about 10 times more frequent than calls and add the product to the number of messages. We define a given edge as “asymmetric” in the behavioral network when the source node is an initiator of communication at least 20% more often than the destination node. Finally, we measure the percentage of asymmetric communication for nodes connected by both asymmetric and symmetric cognitive edges.

Table 5 shows the results of this analysis. We observe that nodes connected by symmetric cognitive edges have close to equal bi-directional communication in the behavioral network. In the first semester only 3% of the symmetric cognitive edges have corresponding behavioral edges asymmetric according to criterion define above. In comparison, asymmetric cognitive edges are much more likely to be asymmetric: In the first semester, asymmetric cognitive edges were about ten times more likely (31%) to feature imbalanced communication than the nodes connected by symmetric cognitive edges. .

Table 5: Difference in communication volume between nodes connected by asymmetric and symmetric cognitive edges.

Semester No.	Asymmetric edges			Symmetric edges		
	No. Calls	No. Messages	% of OSC	No. Calls	No. Messages	% of OSC
Semester 1	69	472	31	58	842	3
Semester 2	25	638	30	39	636	1
Semester 3	40	351	39	112	2038	10
Semester 4	10	256	31	70	1406	6

4.6.4 Communication behavior profile: the “asymmetric sender” profile

We classify nodes that are more likely to be involved in asymmetric communication as *asymmetric senders*. We then examine the communication behavior profile of these nodes to see if the asymmetric sender profile differs from symmetric sender profile. The goal is to verify if nodes with different communication profiles have different characteristics of their cognitive edges. We call students who initiate many asymmetric communications asymmetric senders as we expect them to be more likely to change their friends, given the well known psychological aversion to lack of reciprocity that has been demonstrated in the literature [11]. We find support for the hypothesis in the observation that asymmetric senders retain 7%, 16% and 38% of their friends, while balanced senders retain 25%, 50% and 88% of their friends in the succeeding semesters.

5 Conclusion

In this paper, we study the co-evolution in time of two networks defined by the NetSense data and observe that both networks influence each other temporally. We observe that formation of an edge in the behavioral network is associated with successive formation of a corresponding edge in the cognitive network. We also observe that dissolution of a cognitive edge is often associated with dissolution of its corresponding behavioral edge in the successive semester. So we conclude that both networks affect each other. We also investigate asymmetric cognitive edges, and conclude that the nodes they connect lower communication volume exchange, and lower survival probability than symmetric friendship edges. Moreover, asymmetric cognitive edges are more likely to have corresponding behavioral edges also asymmetric.

Acknowledgements We would like to thank Stephan Dipple for discussions. This work was supported in part by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053 (the Network Science CTA), by the Office of Naval Research (ONR) grant no. N00014-15-1-2640, by the European Commission under the 7th Framework Programme, Agreement Number 316097, and by the Polish National Science Centre, the decision no. DEC-2013/09/B/ST6/02317. The views and conclusions contained in this document are those of the authors.

References

- [1] Baeza-Yates, R.A., Ribeiro-Neto, B.: *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1999)
- [2] Carley, K.M.: Group stability: A socio-cognitive approach. *Advances in Group Processes* **7**, 1–44 (1990)
- [3] Carley, K.M., Krackhardt, D.: Cognitive inconsistencies and non-symmetric friendship. *Social Networks* **18**(1), 1 – 27 (1996). DOI [http://dx.doi.org/10.1016/0378-8733\(95\)00252-9](http://dx.doi.org/10.1016/0378-8733(95)00252-9). URL <http://www.sciencedirect.com/science/article/pii/0378873395002529>
- [4] Chen, M., Bahulkar, A., Kuzmin, K., Szymanski, B.K.: Improving network community structure with link prediction ranking. In: *Proceedings of the 7th Workshop on Complex Networks, CompleNet (2016, to appear)*
- [5] Dong, W., Lepri, B., Pentland, A.S.: Modeling the co-evolution of behaviors and social relationships using mobile phone data. In: *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia, MUM '11*, pp. 134–143. ACM, New York, NY, USA (2011). DOI 10.1145/2107596.2107613. URL <http://doi.acm.org/10.1145/2107596.2107613>
- [6] Hammer, M.: Implications of behavioral and cognitive reciprocity in social network data. *Social Networks* **7**(2), 189 – 201 (1985). DOI [http://dx.doi.org/10.1016/0378-8733\(85\)90005-X](http://dx.doi.org/10.1016/0378-8733(85)90005-X). URL <http://www.sciencedirect.com/science/article/pii/037887338590005X>
- [7] Lazer, D.: The co-evolution of individual and network. *Journal of Mathematical Sociology* **25**(1), 69–108 (2001)
- [8] Miritello, G., Moro, E., Lara, R., Martínez-López, R., Belchamber, J., Roberts, S.G., Dunbar, R.I.: Time as a limited resource: Communication strategy in mobile phone networks. *Social Networks* **35**(1), 89–95 (2013)
- [9] Skyrms, B., Pemantle, R.: A dynamic model of social network formation. In: *Adaptive networks*, pp. 231–251. Springer (2009)
- [10] Striegel, A., Liu, S., Meng, L., Poellabauer, C., Hachen, D., Lizardo, O.: Lessons learned from the netsense smartphone study. In: *Proceedings of the 5th ACM Workshop on HotPlanet, HotPlanet '13*, pp. 51–56. ACM, New York, NY, USA (2013). DOI 10.1145/2491159.2491171. URL <http://doi.acm.org/10.1145/2491159.2491171>
- [11] Wang, C., Lizardo, O., Hachen, D., Strathman, A., Toroczka, Z., Chawla, N.V.: A dyadic reciprocity index for repeated interaction networks. *Network Science* **1**(01), 31–48 (2013)
- [12] Zheleva, E., Sharara, H., Getoor, L.: Co-evolution of social and affiliation networks. In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1007–1016. ACM (2009)

Part V
Diffusion, Epidemics and Spreading
Processes

The spread of ideas in a weighted threshold network

Scott Cox, K.J. Horadam and Asha Rao

Abstract A threshold network is a type of complex network that is useful to model the way in which ideas travel through a human population. Each node has a threshold and only activates if it receives a number of inputs equal to or above the threshold. We build upon work that uses simple distributions of degrees and thresholds by introducing a weighting factor that assigns edges to nodes based on distance apart and similarity of thresholds. This models the way in which people tend to associate more with people of similar beliefs and those who live closer geographically. The model we develop agrees with simulations when the standard deviation of the threshold distribution is low.

1 Introduction

Threshold models of networks have been used to study the spread of rumours through a population [5, 10, 12, 18]. Past research has not fully considered that people with similar ideas and who live close together are more likely to influence each other. In this paper we present a threshold method for predicting the spread of ideas through a human population with these considerations in mind. We then test our model against simulations which show that the model we construct matches simulation results if the standard deviation of the threshold distribution is low.

Disease transmission has been widely studied and the spread of an idea could be seen as similar to the spread of a disease. Consequently the models used to study the spread of diseases can have some efficacy in studying the spread of ideas. Cellular automata models such as in [3, 8] divide a population into a set of cells. These cells are classified as either healthy (susceptible), infected or recovered. After each time step, there is a probability that an infected cell passes on its infection to adjacent cells. In Markov chain models such as in Gomez et al [6] the state of the system at

Scott Cox (e-mail: S9800655@student.rmit.edu.au) · K.J. Horadam (e-mail: kathy.horadam@rmit.edu.au)✉ · Asha Rao (e-mail: asha@rmit.edu.au)✉
RMIT University, Melbourne 3001, Australia

any time only depends on the state of the system in the previous time step. Markov models use a probability that a node will infect an adjacent node and whilst the calculation of these probabilities can become quite complex for large networks, the premise is that an infection spreads by direct contact between two infected nodes with a certain probability. Models based on differential equations such as [14] treat infection in the same way as the above approaches, that is, infection happens from direct contact between individuals and the chance of infection is determined as a probability. Differential equation models give results as to the percentage of the population that is infected.

A model proposed in [15] does use longer connections, connecting nodes which are not physically close. This better represents transmission of ideas, which may be transmitted via phone or internet rather than the disease models which insist on nodes being physically close. A more recent model using differential equations has been proposed in [19] using nodes with a position determined by geographic data to model the spread of rumours in Beijing. The major difficulty in using these models to simulate the spread of ideas is that they use a probability of an infection spreading to an adjacent node each time step. These models do not incorporate the fact that some people are more susceptible to an idea than others.

Ultimately, models of networks which use a threshold are most likely to be of use in the study of the spread of ideas. Watts [18], Gleeson et al [5] and Miller [12] use the idea that a node in a network will only become active once it receives a certain number of inputs, an idea first proposed by Schelling [16] and Granovetter [7] and usually referred to as the Watts threshold model. Hence a node which receives at least as many inputs as its threshold becomes active.

The question as to whether there will emerge a global spreading process, a cascade, is one of the central problems in threshold models. Watts [18] and Miller [12] as well as many other authors [1, 4, 5] are focused on them. This paper is a preliminary study of the effect of making more realistic assignments of thresholds than in [12, 18] and later it is intended to investigate the effects of these on the global emergence of the spreading process. Neither [5], [12] or [18] perform as extensive simulation testing of their model over a wide range of threshold distributions as we do here.

The Watts threshold model can be modified by adding weightings between nodes that depend on how similar their thresholds are and how physically close they are. This represents the idea that similar people who live close to each other are more likely to be friends and communicate with each other, and have a similar susceptibility to a new idea. There has been some work on this problem and we use the work in [9, 17] as motivation to extend this idea. The contribution we make in this paper is an extension of the model in [12] and [5] in order to make it more realistic by adding this weighting and we present the results of simulations to support this.

We start by introducing the model in [12] which uses simple distributions of degree and thresholds. In Section 2.2 we expand this model by adding a weighting factor that assigns edges based on how physically close nodes are and how similar their thresholds are. This represents ideas being more likely to be spread between people who live close together and who have similar viewpoints. The model is tested against simulations in Section 3 and our results are presented.

2 An expanded model

The model in [12] and [5] is a recent model of a threshold network. It predicts the spread of an idea through a network from a simple distribution of thresholds. We describe the model [12] in Section 2.1 and our weighted model in Section 2.2.

2.1 Initial model

We begin by listing some of the equations and assumptions that Miller [12] uses. The discrete time equations that describe the spread of the idea are

$$Q(t) = \sum_k \sum_{r>0} \sum_{m=0}^{r-1} P_u(k, r) \binom{k}{m} \theta(t)^{k-m} (1 - \theta(t))^m, \tag{1}$$

$$\theta(t) = \phi_Q(t - 1), \tag{2}$$

$$\langle K \rangle = \sum_k \sum_r k P_u(k, r), \tag{3}$$

and

$$P_v(k, r) = k P_u(k, r) / \langle K \rangle. \tag{4}$$

The variables are $P_u(k, r)$ which is the probability that a random test node u has degree k and threshold r , $P_v(k, r)$ is the probability that a random neighbour v of the test node has degree k and threshold r , $Q(t)$ is the probability a test node u is still inactive at time t , $\phi_Q(t)$ is the probability that a random neighbour of a test node u is still inactive at time t given that the test node u is inactive, $\theta(t)$ is the probability that a random neighbour has not transmitted to the test node at time t and $\langle K \rangle$ is the average degree of nodes in the population.

It is assumed that we can pick any node as the test node. Equation 1 is the sum of the probabilities that the test node receives fewer signals from its neighbours than its threshold. Equation 2 states that the probability of a test node being inactive is the probability of a random neighbour being inactive in the previous timestep. Since we can pick any node as the test node and calculate the probability that a neighbour has transmitted, $\theta(t)$ is the proportion of inactive nodes in the network at time t . Another assumption is that a neighbour of a test node has $k/\langle K \rangle$ more neighbours than the test node. This assumption is based on a neighbouring node being likely to have more edges than any randomly chosen test node.

2.2 Improved model

Here we introduce our improvements to the model in [12].

2.2.1 Threshold distribution

The threshold r of a node is the number of neighbours that must be active in order for a node to change from inactive to active. In the examples in [12], Miller gives most nodes a threshold of 2 apart from a small population of initially active, threshold 0 nodes. This distribution of thresholds is unlikely to match those in a real human population. Here we make the assumption that a normal distribution will model susceptibility of humans to new ideas reasonably well and better than a constant distribution. There is already some work that addresses heterogenous threshold distributions [11] and we build upon this.

In all of the cases we examine, the thresholds are integer values between 0 and 9. We choose this range of thresholds as a starting point because results in [2] based on human psychology suggest that this is a reasonable range of values. If we take the probabilities $P'(r)$ for $r \in \{0, 1, 2, \dots, 9\}$ from a normal distribution $N(\mu, \sigma)$ then $\sum_{r=0}^9 P'(r) \neq 1$. We normalise to obtain the probability $P(r)$ of a node having threshold r :

$$P(r) = P'(r) / \sum_{r=0}^9 P'(r) \quad (5)$$

An example for $N(5, 1.5)$ appears in Figure 1.

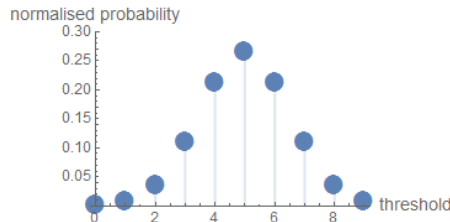


Fig. 1: The normalised distribution of thresholds for $N(5, 1.5)$. Each dot represents the probability of a certain threshold being assigned to a node.

2.2.2 Geographical proximity

In [12], Miller uses nodes with degrees equally distributed between 2,4 and 6. This is unlikely to be realistic as geographical proximity of nodes is likely to be important. People are likely to have meaningful conversations with people who they live near and can speak with face to face. It is true that current technology allows long distance communication and we also include the possibility of some longer distance links forming.

Here we use a simple square grid of nodes to minimise the complexity of calculations, but other spatial distributions are obviously possible and other topologies could match real world situations more closely. The nodes are the intersection points

of an $N \times N$ grid. Each of the N^2 nodes is assigned a threshold by sampling from the discrete distribution of Section 2.2.1.

The distance between nodes is measured using the Chebyshev metric

$$D_\infty(d, d') = \lim_{j \rightarrow \infty} \left(\sum_{i=1}^2 |d_i - d'_i|^j \right)^{1/j} \quad (6)$$

$$= \max\{|d_1 - d'_1|, |d_2 - d'_2|\}.$$

2.2.3 Weighting factor and calculation of degrees

In contrast to Miller's model, in which thresholds and degrees are assigned to each node, we use a weighting factor based on similarity of threshold and distance to neighbour nodes in order to assign degrees. We assume that a node is more likely to connect to a node in close proximity. Threshold is also important since we assume people of similar opinions are more likely to be associated with each other than people of differing opinions.

The weighting factor we use for the weight of a link between the test node and another node is

$$w(r, r', d, d') = \frac{|N/2 - D_\infty(d, d')|}{N/2} \left(1 - \frac{|r - r'|}{r_{\max}} \right) \frac{1}{a}, \quad (7)$$

where r = threshold of the test node, r' = threshold of a different node, d = position of the test node, d' = position of the different node and a is a parameter we can vary to control the average degree. Thus $w \in [0, 1/a]$. This weighting factor allows some edges between distant nodes to form. This is a type of small-world effect. We allow some nodes to have $r > k$, this represents people who refuse to change their opinion under any circumstances. This causes our definition of a threshold to differ somewhat from the definition in [5, 18].

Our expected value of the degree of a node at position d and threshold r is now

$$k(r, d) = \sum_{r'=0}^9 \sum_{d' \neq d} w(r, r', d, d') P_V(r'). \quad (8)$$

This equation adds the probabilities that a test node and each other node are connected. The value of k in Equation 8 is always rounded to the nearest integer. The expected degree $k(d)$ for a node at d is

$$k(d) = \sum_{r=0}^9 k(r, d). \quad (9)$$

2.2.4 Equations governing spread of ideas

We now incorporate the weighting factor and normal distribution of thresholds into Miller’s model in [12]. Thus Equation 2 becomes

$$\theta(t) = \sum_d \sum_{r>0} \sum_{m=0}^{r-1} \frac{k(r,d)P_u(r)(1/N^2)}{\langle K \rangle} \binom{k(r,d)-1}{m} \theta(t-1)^{k(r,d)-1-m} (1-\theta(t-1))^m. \tag{10}$$

and Equation 3 transforms into

$$\langle K \rangle = \frac{1}{N^2} \sum_d k(d).$$

In what follows, the test node u is always the node at the centre of the grid at $d^* = (N/2, N/2)$. So averaging over d is unnecessary and Equation 10 becomes

$$\theta(t) = \sum_{r>0} \sum_{m=0}^{r-1} \frac{k(r,d^*)P_u(r)}{\langle K \rangle} \binom{k(r,d^*)-1}{m} \theta(t-1)^{k(r,d^*)-1-m} (1-\theta(t-1))^m \tag{11}$$

and $\langle K \rangle$ becomes

$$\langle K \rangle = k(d^*). \tag{12}$$

3 Results and Discussion

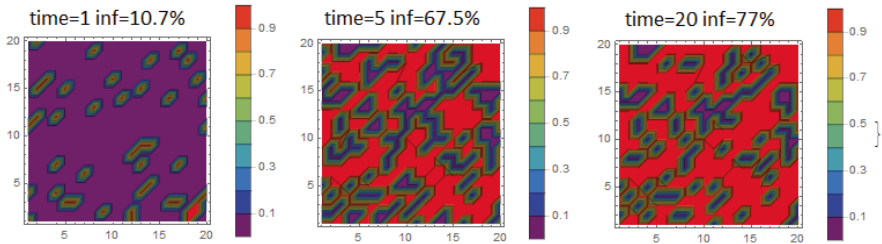


Fig. 2: Simulation results for a 20×20 grid with a threshold distribution $N(2, 1.5)$

We test our new model against a simulation written in Mathematica. The simulation functions as follows:

Algorithm 13 Simulation code

```

1: Input  $N, \mu, \sigma, m, T$ 
2: Generate an  $N \times N$  grid of nodes
3: for  $i = 1$  to  $m$  do
4:   Assign each node a threshold using the threshold distribution
5:   For each node, selected at random without replacement, generate a random
   number
   between 0 and 1. For any remaining node, if the random number is less than
    $w(r, r', d, d')$ 
   then the two nodes are connected by an edge.
6:   for  $t = 1$  to  $T$  do
7:     An active node sends a signal to each node it is connected with.
8:     Each node that receives a number of signals equal to or exceeding its
   threshold in
   this timestep now activates.
9:   end for
10:  Store the result.
11: end for
12: The mean and standard deviation of  $m$  results is calculated.

```

The simulation will test whether our equations accurately describe the spread of ideas through the weighted threshold network we use. Since the simulation gives a different result each time it is run, we give the mean and standard deviation of the result of m outputs of the simulation. Each instance runs T times until all nodes are active or a stable percentage of active nodes is reached.

Example 3.1. We use $N(2, 1.5)$ as a threshold distribution, giving $Pr(0) = 0.1145$. So our initially inactive proportion of the population is $1 - 0.1145 = 0.8855$. We choose $N = 20$ and $a = 25$ and use Equations 7 and 12 to calculate $\langle K \rangle = 4.71$. Equation 11 gives $\theta(1) = 0.791$, $\theta(5) = 0.39888$ and $\theta(20) = 0.173861$.

An example of the simulation output is shown in Figure 2.

If the simulation is run 100 times, we have a mean infection of 83.36% and a standard deviation of 2.97% after 20 time steps. This agrees well with Equation 11. The simulation predicts 83.36% and Equation 11 predicts $1 - \theta(20) = 82.61\%$. The large seed size we used led to an almost global spread.

The first simulations we run involve a threshold distribution with a small standard deviation of $\sigma = 1.5$. Table 1 compares the simulation results with the result of Equation 11. Here $T = 20$ except for $N(3, 1.5)$, $N = 40$ where we use $T = 24$. For this result, Equation 11 needed more time to converge to a final value. There is some considerable disagreement here. The standard deviation for the $N(3, 1.5)$, $N = 20$ result is especially high since the threshold distribution is such that there is an equally likely chance of the grid being populated by low threshold nodes with $r < 3$ as it is

Table 1: Comparison of $1 - \theta(T)$ from simulation and Equation 11.

N	a	m	$P(0)\%$	threshold distribution	$\langle K \rangle_{theory}$	$\langle K \rangle_{simulation}$	theory %	simulation %
20	25	100	0.10	$N(5, 1.5)$	4.23	4.86 ± 0.12	0.049	0.14 ± 0.20
20	25	100	0.76	$N(4, 1.5)$	4.25	4.84 ± 0.11	0.6	0.91 ± 0.59
20	25	100	3.63	$N(3, 1.5)$	4.47	4.87 ± 0.11	4.71	15.83 ± 12.54
20	25	100	11.45	$N(2, 1.5)$	4.71	4.94 ± 0.12	82.6	83.83 ± 2.94
40	60	20	0.10	$N(5, 1.5)$	7.16	8.10 ± 0.10	0.059	0.08 ± 0.08
40	60	20	0.76	$N(4, 1.5)$	7.16	8.13 ± 0.06	0.632	1.25 ± 0.54
40	60	20	3.63	$N(3, 1.5)$	7.39	8.16 ± 0.08	96.27	94.57 ± 0.76
40	60	20	11.45	$N(2, 1.5)$	6.96	7.89 ± 0.09	99.27	97.68 ± 0.26

by high threshold nodes with $r > 3$. The standard deviation for the $N(3, 1.5)$, $N = 40$ result is lower since the value for $\langle K \rangle$ is high enough to ensure that there is enough connectivity that the number of nodes with $r > 3$ does not stop ideas flowing.

We now do the same for threshold distributions with a different standard deviation and present the results in Table 2. The larger standard deviation gives a greater disagreement between simulation and theory. This is because we round the value for k in Equation 11. When the standard deviation of the threshold distribution is small, most of the thresholds are similar and we have a larger value for k . This will result in larger values of k being rounded off to the nearest integer in Equation 11 and therefore a smaller rounding error. However, integer values are required for the binomial term in Equation 11.

As is expected, the greater the initial concentration of infected people $P(0)$, the greater the value of $1 - \theta(T)$. The value for $\langle K \rangle$ is consistently higher in the

Table 2: Comparison of $1 - \theta(T)$ from simulation and Equation 11.

N	a	m	$P(0)\%$	threshold distribution	$\langle K \rangle_{theory}$	$\langle K \rangle_{simulation}$	theory %	simulation %
20	25	100	4.27	$N(6, 4)$	3.68	4.02 ± 0.11	10.57	6.45 ± 1.98
20	25	100	5.81	$N(5, 4)$	3.59	4.03 ± 0.11	5.57	10.55 ± 2.59
20	25	100	7.69	$N(4, 4)$	3.59	3.98 ± 0.11	7.85	15.85 ± 3.73
20	25	100	9.90	$N(3, 4)$	3.68	4.00 ± 0.11	13.15	24.51 ± 4.98
40	60	20	4.27	$N(6, 4)$	6.13	6.74 ± 0.07	3.54	10.98 ± 2.31
40	60	20	5.81	$N(5, 4)$	6.01	6.68 ± 0.06	5.41	21.00 ± 2.77
40	60	20	7.69	$N(4, 4)$	6.01	6.68 ± 0.07	10.57	43.89 ± 4.63
40	60	20	9.90	$N(3, 4)$	6.13	6.75 ± 0.07	22.88	59.07 ± 3.78

simulation than it is when using Equation 12. This is because $\langle K \rangle$, in Equation 12, is calculated as the expected number of edges that the central node has rather than the total number of edges in the network divided by the total number of nodes.

3.1 Weighted average to speed up computation

We can speed up computation time considerably by taking a weighted average of $\frac{|N/2 - D_\infty(d^*, d')|}{N/2}$ by noting that most of the nodes are in the outer regions of the grid. Since there are $8n$, $n \geq 1$ nodes in each unit of Chebyshev distance n from the centre, we have

$$\begin{aligned} \text{Weighted average} &= \left(\sum_{n=1}^{N/2-1} 8n \times \frac{|N/2 - n|}{N/2} \times \frac{1}{N^2 - 1} \right) \times N^2 \\ &= \frac{N^2(N^2 - 4)}{3(N^2 - 1)} \end{aligned} \tag{13}$$

This gives us

$$W(r, r') = \frac{N^2(N^2 - 4)}{3(N^2 - 1)} \left(1 - \frac{|r - r'|}{r_{\max}} \right) \frac{1}{a} \tag{14}$$

We have thus defined $W(r, r')$ as a weighting factor that just depends on the threshold r of the test node, the threshold r' of a neighbour node and the parameter a used to control how many edges each node has. This allows us to define an expected value for the degree of a node that does not sum over all of the nodes in the grid. We call this function $K(r, r') = W(r, r')P(r')P(r)$ and we note that we replace $w(r, r', d, d')$ in Equation 7 by the $W(r, r')$ we just defined.

This significantly speeds up our calculations and simplifies them by removing the position terms d and d' . Rather than summing over $100(N^2 - 1)$ terms, we now sum over 100 terms. The results using this weighted average differ very little from the results using the more time consuming calculation. We compare results from a 100×100 grid with a threshold normal distribution $N(3, 1.5)$ and a value of a in the weighting equation (Equation 7) of 120 in Table 3 below.

Table 3: comparison of results using weighted average for a 100×100 grid with a threshold distribution $N(3, 1.5)$.

time	$\theta_{\text{not weighted}}(t)$	$\theta_{\text{weighted}}(t)$
$\theta(0)$	0.963689	0.963689
$\theta(1)$	0.86026	0.851148
$\theta(2)$	0.402703	0.37095
$\theta(3)$	0.00118539	0.000803752
$\theta(4)$	0.0000357656	0.0000357656

4 Conclusion and Future work

We have extended the model presented in [12] and see that by adding weighted connections that take into account that people with similar ideas and who live close together are more likely to influence each other, we have a model which agrees with simulations as long as we use threshold distributions with a low standard deviation. The way that the model timesteps have been calculated, especially the rounding of k in Equation 11 causes most of the disagreement.

Future work may involve using a weighting factor such as

$$w(r, r', d, d') = \frac{|N - D_{\infty}(d, d')|}{N} \left(1 - \frac{|r - r'|}{r_{\max}} \right) \frac{1}{a}, \quad (15)$$

which decreases with distance and has no small-world effect. We can also allow any node to be the test node and use Equation 10 together with a weighted average. This will allow faster computation time and a more general result. Different arrangements of nodes other than a grid could be studied and the conditions in which a cascade occurs investigated. Rather than limiting thresholds to the range 0 to 9, varying this range and perhaps making it vary over $[0, k]$ may produce some interesting results.

Acknowledgement : This work forms part of the PhD thesis of the first author taken under the supervision of the second and third authors. We thank the reviewers for their many helpful comments.

References

- [1] Centola, Damon, Eguíluz, Víctor M and Macy, Michael W. Cascade dynamics of complex propagation. *Physica A: Statistical Mechanics and its Applications*. **374**. 449–456 (2007)
- [2] DiFonzo, Nicholas, Beckstead, Jason W, Stupak, Noah and Walders, Kate Validity judgments of rumors heard multiple times: the shape of the truth effect. *Social Influence*. **11**. 22–39 (2016)
- [3] Fuentes, M. and Kuperman, M. Cellular automata and epidemiological models with spatial dependence. *Physica A: Statistical Mechanics and its Applications*. **267**. 471–486 (1999)
- [4] Gai, P and Kapadia, S. Contagion in financial networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*. **466**. 2401–2423 (2010)

- [5] Gleeson, James P. and Cahalane, Diarmuid J. Seed size strongly affects cascades on random networks. *Phys Rev E Stat Nonlin Soft Matter Phys.* **75**. 056103 (2007)
- [6] Gomez, S., Arenas, A., Borge-Holthoefer, J., Meloni, S. and Moreno, Y. Discrete-time Markov chain approach to contact-based disease spreading in complex networks. *Europhysics Letters.* **89**. 38009 (2010)
- [7] Granovetter, M. Threshold Models of Collective Behavior. *American Journal of Sociology.* **83**. 1420–1423 (1978)
- [8] Hawkins, J.M. and Molinek, D.K. Markov cellular automata models for chronic disease progression. *International Journal of Biomathematics.* **8**. 1550085 (2015)
- [9] Hurd, T.R. and Gleeson, J.P. On Watts' cascade model with random link weights. *Journal of Complex Networks.* **1**. 25–43 (2013)
- [10] Karimi, Fariba and Holme, Petter Threshold model of cascades in empirical temporal networks. *Physica A: Statistical Mechanics and its Applications.* **392**. 3476–3483 (2013)
- [11] Karsai, M., Iniguez, G., Kikas, R., Kaski, K. and Kertesz, J. Local cascades induced global contagion: How heterogeneous thresholds, exogenous effects, and unconcerned behaviour govern online adoption spreading. *Sci Rep.* **6**. 27178 (2016)
- [12] Miller, Joel. Complex contagions and hybrid phase transitions. *Journal of Complex Networks.* **4**. 1–23 (2015)
- [13] Newman, M., Strogatz, S. and Watts, D. Random graphs with arbitrary degree distributions and their applications. *Phys Rev E Stat Nonlin Soft Matter Phys.* **64**. 026118 (2001)
- [14] Rao, S. and Kumar, N. A dynamic model for infectious diseases: The role of vaccination and treatment. *Chaos, Solitons & Fractals.* **75**. 34–49 (2015)
- [15] Sander, L.M., Warren, C.P., Sokolov, I.M., Simon, C. and Koopman, J. Percolation on heterogeneous networks as a model for epidemics. *Mathematical Biosciences.* **180**. 293–305 (2001)
- [16] Schelling, T. Dynamic Models of Segregation. *Journal of Mathematical Sociology.* **1**. 143–186 (1971)
- [17] Toole, J.L., Cha, M. and Gonzalez, M. Modeling the adoption of innovations in the presence of geographic and media influences. *PloS one.* **7**. e29528 (2012)
- [18] Watts, D.J. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences.* **99**. 5766–5771 (2002)
- [19] Zhang, N., Huang, H., Duarte, M. and Zhang, J. Risk analysis for rumor propagation in metropolises based on improved 8-state ICSAR model and dynamic personal activity trajectories. *Physica A: Statistical Mechanics and its Applications.* **451**. 403–419 (2016)

Information Diffusion in Heterogeneous Groups

Jennifer M. Larson

Abstract Standard approaches to the study of information diffusion draw on analogies to the transmission of diseases or computer viruses, and find that adding more random ties to a network increases the speed of information propagation through it. However, a person sharing information in a social network differs from a computer transmitting a virus in two important respects: a person may not have the *opportunity* to pass the information to every tie, and may be *unwilling* to pass the information to certain ties even when presented with the opportunity. Accounting for these two features reveals that, while additional random ties allow information to jump to distant regions of a network, they also change the composition of network neighborhoods. When the latter increases the proportion of neighbors to whom people are less willing to pass information, the result can be a net decrease in diffusion. I show that this is the case in heterogeneous, homophilous networks: the addition of random ties strictly impedes information dissemination, and the impediment is increasing in both original homophily and the number of new ties.

1 Introduction

The study of information diffusion in social systems applies insights from epidemiology to the spread of ideas, innovations, or behavior from node to node in a social network [1, 11, 20, 23, 24, 26]. The basic logic holds that nodes “infected” with an idea or behavior are “contagious”; network neighbors of the infected are exposed and hence susceptible to the infection, with variants accounting for the consequences of exposure to multiple sources [4, 5], variation in motivation [7, 10], the cumulative effect of repeated exposures [8, 9], and homophily with respect to susceptibility [6].

The analogy to disease spread has generated important findings about the relationship between network structure and information diffusion. Increasing the proportion

Jennifer M. Larson (e-mail: jenn.larson@nyu.edu)✉
Department of Politics, New York University, 19 w. 4th St. New York, NY 10012

of random ties in a regular network dramatically increases the propagation rate of cascades [12, 19], the presence of particularly well-connected nodes is beneficial for diffusion [16, 20, 21, 25], and random rewiring in small world networks accelerates diffusion [12, 19]. In general, adding random ties to a network will improve diffusion.

While the epidemiological approach has offered valuable insights, ties in a social network function quite differently for the spread of information than ties in a contact network function for the spread of a disease. In the case of a contact network, a tie by definition makes an alter susceptible to the disease of the ego. In the case of a social network, a tie does not *by definition* spread information to an alter. A tie indicates a social relationship. Whether or not this social relationship results in an ego passing information to an alter depends on a variety of factors: whether the two happen to encounter each other while the information is salient, whether they are together for long enough for the information to come up, whether the ego thinks the information is relevant to the alter, whether the ego is willing to share with the particular alter, and so on.

In fact, for the type of information that is often the subject of diffusion studies, an ego may have good reason to prefer to share it with some social ties over others. In the case of collective action, the information may be a person's dissatisfaction with a regime or her willingness to participate in a protest [5, 7]. Given the sensitivity of this information, especially in oppressive regimes, a person may only be willing to pass it to her most trusted social ties. In the case of technology adoption, especially in the developing world, relevant information may be news of a development organization offering startup loans or handing out new technology like fertilizer [2]. A person may judge the opportunity to be finite or selectively beneficial and prefer to share information of it with her social ties that are kin or members of her salient in-group like her tribe [14]. In social networks, a person can choose whether to share information or whether to withhold it on a tie-by-tie basis.

In this conceptualization of information diffusion, a person in a social network will only spread information to a particular network neighbor if (1) she is presented with an opportunity to do so, and (2) is willing to share the information with that neighbor.

I account for these two features in a model in which a person has a finite number of opportunities to spread information with network neighbors. Individuals in the network have a type, which could represent ethnicity, tribe, political party, or any other salient division correlated with willingness to share new information. Given an opportunity, a person always shares information with a same-type neighbor but occasionally withholds information from a different-type neighbor.

When only one type is present in the network, the results reproduce those of earlier work: the addition of random ties allows information to jump to distant regions of the network, increasing the speed of diffusion. When multiple types are present, however, random ties introduce a second effect: they change the composition of network neighborhoods, possibly increasing the chances that the limited number of encounters will be with different-type neighbors. I show that in heterogeneous networks with type-homophily, the addition of random ties can result in the second effect dominating. In heterogeneous networks, the addition of random ties can strictly

reduce the speed of information diffusion. The reduction is increasing in the original homophily, the number of types in the network, and the number of added ties.

Since homophilous communities within a network would facilitate information spread, these results are consistent with others' findings that network modularity can improve information dissemination via social reinforcement [3, 18]. However, the result here is even stronger: not only would rearranging links to reduce modularity impede information spread, but adding *new* links to the network at random can strictly impede information spread as well.

These findings refine those of earlier work, showing that the benefit of additional random ties hinges on plentiful opportunities to share information with all network neighbors and perfect willingness to share the information at every opportunity. In the more realistic case of limited opportunities and differential willingness to share, the addition of random ties may be counterproductive. In heterogeneous groups, the greater the type-homophily, the more damaging random ties are to the wide reach of information.

2 An Opportunity Model of Information Diffusion

Suppose a network g is comprised of a finite number of nodes that each have one of n types $\tau \in \{\tau_1, \dots, \tau_n\}$. A type is a descriptive feature of a node and is used to separate an in-group from out-groups, like membership in a certain tribe or political party. Call a network *homogeneous* if $n = 1$; that is, if all nodes have the same type. A network is *heterogeneous* if $n > 1$.

Consider a simple model of information diffusion over time in which individuals may pass along new information to some network neighbors when presented with the opportunity. Call i 's neighbors in g $N_i(g)$. In the model, an individual's willingness to share information depends on type: she is more willing to share information with same-type nodes than with different-type nodes. Specifically, the diffusion process proceeds as follows:

- $t = 0$ One node i is randomly selected and endowed with information.
- $t = 1$ Seed i randomly encounters x of her network neighbors, $N_i(g)$. In each encounter, she passes information to the neighbor with probability p_{same} if she and the neighbor are both the same type, and probability $p_{dif} < p_{same}$ if they are different types.
- $t = 2$ All j who learned information in $t = 1$ randomly encounter x of their neighbors, $N_j(g)$, passing information with probabilities p_{same} and p_{dif} .
- ... Repeats for all who learned information in the previous period until the information has reached everyone in the network or the spread halts.

2.1 Consequences of randomly added links

Randomly added or rewired ties have been found to improve information diffusion in homogeneous networks because random ties allow information to “jump” to distant network locations [12, 19]. However, the diffusion process specified in section 2 introduces a second, potentially-competing effect in heterogeneous networks. Randomly added ties can change the composition of nodes’ neighborhoods. If neighborhoods are comprised of more ties to other-type nodes, the expected number of neighbors who receive the information declines.

Dual Effects of Random Ties in Heterogeneous Networks

Jump effect: random ties allow information to jump across distant network locations, improving information dissemination.

Composition effect: random ties change the composition of a node’s neighborhood, potentially impeding information dissemination.

In a heterogeneous network, which effect dominates— the jump effect which improves dissemination or the composition effect which hinders dissemination— depends on the relationship between homophily and the distribution of types in the network.

Node i ’s network neighborhood $N_i(g)$ can be decomposed into $N_i^{same}(g)$, the subset of his network neighbors that are the same type as i , and $N_i^{dif}(g)$, the subset that are different. The expected number of nodes who receive information from i can then be written

$$\frac{x}{\#N_i(g)} \left(\#N_i^{same}(g)p^{same} + \#N_i^{dif}(g)p^{dif} \right), \tag{1}$$

where # indicates the cardinality of a set.

The consequences of an additional tie added at random will depend on the proportion of the nodes in g that are the same type as i . Call q^{τ_k} the proportion of nodes in g that are type τ_k . For simplicity, from any node i ’s perspective, call q_i^{same} the proportion of nodes of i ’s type in g . Now a random link added to $N_i(g)$ will reduce the value of (1) whenever

$$\frac{\#N_i^{same}(g)}{\#N_i(g)} - q_i^{same} > 0. \tag{2}$$

That is, when the network is homophilous with respect to type so that a larger proportion of a node’s neighbors are his same type relative to the frequency of his type in the overall network, the addition of random ties will strictly reduce the expected number of people that that node informs.

The extent to which the expected number of nodes who receive information from i declines depends on the magnitude of the left hand side of (2). The greater the type homophily, the bigger impact random ties will have on reducing the expected number of people that a node informs.

When this relationship is prevalent enough throughout a network, network-wide information dissemination can be strictly impeded by the addition of random ties. The next section demonstrates the aggregate results using a simulated information diffusion process.

3 Simulated Information Spread

In this section I simulate the information diffusion process from Section 2 on simple networks generated with varying levels of homophily, heterogeneity, and random tie additions.

3.1 *The Downside to Density*

I begin by generating four heterogeneous networks, each with two types of nodes. The networks have 234 nodes, half of which are each type, and 864 links. Each network is generated by randomly adding links according to a specified probability of attaching to a same-type node. One network is generated for each same-type node probability $\{.5, .65, .8, .95\}$. Let the difference between the proportion of same-type links present and the proportion of same-type links that would be observed by uniformly random link formation be called the network's "homophily." With two groups of equal size, the expected proportion of random same-type links is $.5$, yielding networks with homophily values $\{0, .15, .3, .45\}$.

I consider the consequences of increasing density for information diffusion by randomly adding links to the network. For each value of homophily, I add links such that the total number of links increases by a factor of 1, 2, 3, and 10.

Table 1 summarizes the interpretation of the model parameters and the values to which they are set in the simulations reported below.

Figure 1 shows the results of the simulated information diffusion process on each of these networks, grouped by homophily value. In each quadrant, the curves plot the average proportion of the network that is informed by the timestep on the horizontal axis over a set of 500 simulations for a particular value of density increase. Since the population is finite, $p_{same} > 0$, and $p_{dif} > 0$, diffusion follows the characteristic s-shape. The lower the curve, the slower the diffusion.¹

When the network exhibits no homophily (top left), randomly adding links can improve information dissemination. In this case, since the composition of the population matches the composition of neighborhoods on average, randomly adding

¹ This represents an impediment to diffusion in the sense that information reaches people more slowly, and also in the sense that by any given point in time, fewer people are informed.

Table 1: Model Parameters

Parameter	Definition	Set to
x	Number of network neighbors a newly-informed node 2 encounters in a period	
p_{same}	Probability pass news to an encountered neighbor if 1 neighbor is same type	
p_{dif}	Probability pass news to an encountered neighbor if .5 neighbor is different type	
τ	= Set of types	$\{\tau_1, \tau_2\}$,
$\{\tau_1, \dots, \tau_n\}$		$\{\tau_1, \tau_2, \tau_3\}$,
		$\{\tau_1, \tau_2, \tau_3, \tau_4\}$
q^{τ_k}	Proportion of type $\tau_k \in \tau = \{\tau_1, \dots, \tau_n\}$ present in the 1/ n network	
Homophily	Proportion same-type ties in network minus propor- tion same-type ties expected under random tie forma- tion	$\{0, .15, .3, .45\}$
Diversity	Number of types, or “groups”, present in the network	$\{2, 3, 4\}$
Density Inc.	Factor by which number of links is increased; e.g. 2 adds 200% of original links as new links	$\{0, 1, 2, 3, 10\}$

links has no composition effect. The jump effect dominates, improving information dissemination on net.

When network neighborhoods contain more same-type links than would be expected based on the overall network composition (exhibit positive homophily), the composition effect is present alongside the jump effect. In the cases of positive homophily shown in Figure 1, the composition effect dominates: an increase in density actually impedes information diffusion. The greater the number of links added, the worse the diffusion.

Note that the number of randomly-added ties is large in these simulations, in some cases increasing the number of links in the network many-fold. Under standard epidemiological models of information diffusion, the improvement in diffusion would be vast. Here, these large additions actually *reduce* the spread of information. Moreover, these simulations assume that individuals share with other-types half of the time ($p_{dif} = .5$). When people are more hesitant to share with other types so that p_{dif} is smaller, the reduction in information spread is even greater.

3.2 The Role of Diversity

Figure 2 holds the probability of same-type links constant and increases the number of equal-sized groups in the network (the network’s “diversity”). The vertical bars

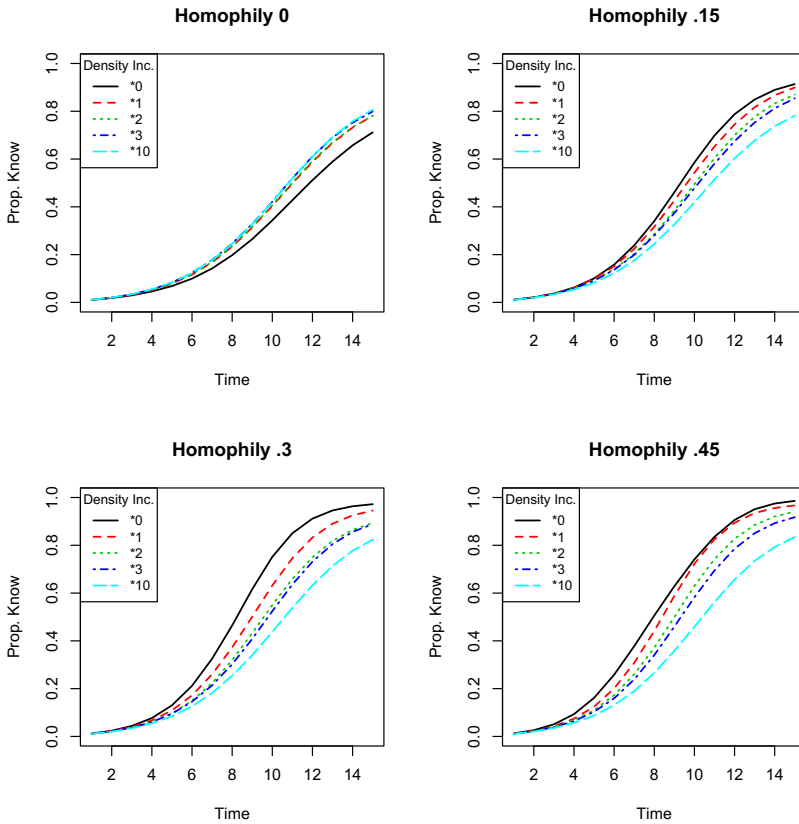
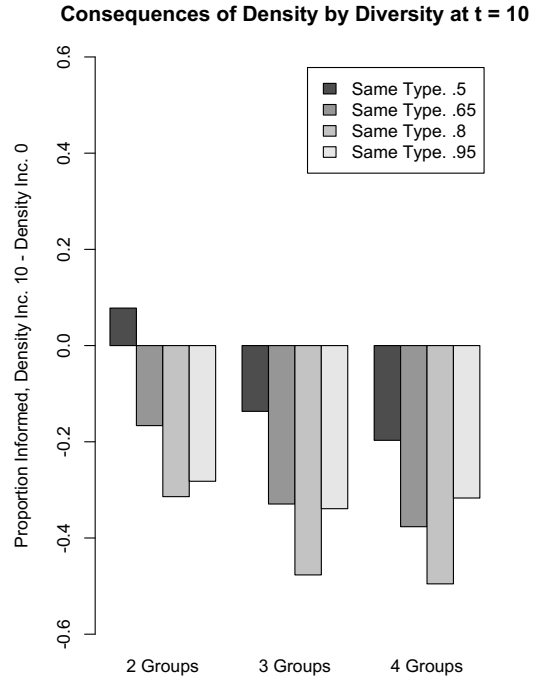


Fig. 1: Proportion of network informed by each timestep in simulated information spread on a network with $\tau = \{\tau_1, \tau_2\}$, and $q^{\tau_1} = q^{\tau_2} = \frac{1}{2}$. Simulation parameters set to $x = 2$, $p_{same} = 1$, and $p_{dif} = .5$. When homophily = 0, random ties will not change neighborhood compositions on average, so the jump effect dominants and increasing density strictly improves information diffusion. At greater values of homophily, increasing density does change neighborhood compositions and strictly impedes information diffusion.

display the proportion of the network that has been informed on average by the tenth timestep of the simulations for each network when it has ten times the number of original links added at random minus this value for the original network. In other words, this displays the gain or loss from increasing the density of each network given a certain number of groups present in the network.

The cluster of bars on the left translates the information from Figure 1 in which there are two groups present in the network. These show that when the probability of sharing a link with a same-type is greater than .5, greater density reduces the

Fig. 2 Difference in proportion of network informed by timestep 10 when the density is increased by a factor of 10 compared to the proportion informed by timestep 10 given the original density. Shown for 2 groups ($\tau = \{\tau_1, \tau_2\}$ with $q^{\tau_1} = q^{\tau_2} = \frac{1}{2}$), 3 groups ($\tau = \{\tau_1, \tau_2, \tau_3\}$ with $q^{\tau_1} = q^{\tau_2} = q^{\tau_3} = \frac{1}{3}$), and 4 groups ($\tau = \{\tau_1, \tau_2, \tau_3, \tau_4\}$ with $q^{\tau_1} = q^{\tau_2} = q^{\tau_3} = q^{\tau_4} = \frac{1}{4}$). Simulation parameters set to $x = 2$, $p_{same} = 1$, and $p_{dif} = .5$. The downside to greater density is more pronounced in more diverse networks.



proportion of the network that has been informed by the tenth time step. The next two sets of clusters show the same from the case where there are three and four types of equal size present in the network, respectively. Comparing across clusters shows that the impediment to diffusion is greater when diversity is higher.

The negative consequences of adding random links to a network are even more acute in the presence of greater diversity.

4 Conclusion

Previous studies have found that the addition of random ties unambiguously improves information dissemination. Additional random ties generate a “jump effect,” allowing information to jump from region to region within networks, speeding the spread of information. However, the present work suggests that there is an additional, at times competing effect that is masked when important features of information-sharing in social networks are unaccounted for.

Ties in social networks represent potential opportunities for the spread of information rather than certain conduits of information. People may be limited in the number of encounters that would permit information-sharing, and people can decide whether

or not to share information with any candidate recipient when given the opportunity. This paper builds these two features into a model of information diffusion by assuming a uniform number of encounters per person and the presence of types such that people are more willing to share information with a same-type than a different-type neighbor.

Accounting for these features reveals that a “composition effect” can result in random ties impeding the spread of information. When random ties reduce the proportion of same-type nodes in nodes’ neighborhoods, opportunities to share information are more likely to arise with people of a different type. Since people are more hesitant to share with different types, random ties can impede overall information dissemination.

Note that the two effects can be on net negative, even when people are still willing to share information with different type ties *some of the time*. In heterogeneous groups, especially ones with high homophily, greater density can actually strictly reduce the speed with which information spreads throughout a network.

In addition to revealing a potentially negative consequence of network density in diverse groups, these results also help make sense of recent empirical findings in the social sciences showing that group composition is directly related to both trust [22] and the reach of novel information [14]. Areas that are heterogeneous in salient types— for instance those that are ethnically diverse— fare poorly in outcomes that require information to spread to coordinate outcomes like providing public goods [17], keeping aspiring rebel groups’ secrets from the government[15], and enforcing behavior through peer sanctions [13]. Heterogeneous groups may face difficulties due to problems with information dissemination that homogeneous groups are able to avoid.

References

- [1] Banerjee, A., Chandrasekhar, A.G., Duflo, E., Jackson, M.O.: The diffusion of microfinance. *Science* **341**(6144) (2013)
- [2] Banerjee, A.V., Chandrasekhar, A., Duflo, E., Jackson, M.O.: Gossip: Identifying central individuals in a social network. Available at SSRN 2425379 (2014)
- [3] Centola, D.: The spread of behavior in an online social network experiment. *science* **329**(5996), 1194–1197 (2010)
- [4] Centola, D., Macy, M.: Complex contagions and the weakness of long ties. *American journal of Sociology* **113**(3), 702–734 (2007)
- [5] Centola, D.M.: Homophily, networks, and critical mass: Solving the start-up problem in large group collective action. *Rationality and society* **25**(1), 3–40 (2013)
- [6] Chiang, Y.S.: Birds of moderately different feathers: Bandwagon dynamics and the threshold heterogeneity of network neighbors. *Journal of Mathematical Sociology* **31**(1), 47–69 (2007)

- [7] Chwe, M.: Communication and Coordination in Social Networks. *Review of Economic Studies* **67**(1), 1–16 (2000)
- [8] Dodds, P.S., Watts, D.J.: Universal behavior in a generalized model of contagion. *Physical review letters* **92**(21), 218,701 (2004)
- [9] Dodds, P.S., Watts, D.J.: A generalized model of social and biological contagion. *Journal of theoretical biology* **232**(4), 587–604 (2005)
- [10] Granovetter, M.: Threshold models of collective behavior. *American journal of sociology* pp. 1420–1443 (1978)
- [11] Jackson, M.O., Rogers, B.W.: Relating network structure to diffusion properties through stochastic dominance. *The BE Journal of Theoretical Economics* **7**(1) (2007)
- [12] Kleinberg, J.: Small-world phenomena and the dynamics of information. *Advances in neural information processing systems* **1**, 431–438 (2002)
- [13] Larson, J.M.: Networks and interethnic cooperation. *Journal of Politics* **Forthcoming** (2017)
- [14] Larson, J.M., Lewis, J.I.: Ethnic networks. *American Journal of Political Science* **Forthcoming** (2017)
- [15] Larson, J.M., Lewis, J.I.: Rumors, kinship networks, and rebel group formation. Working Paper (2016)
- [16] López-Pintado, D.: Diffusion in complex social networks. *Games and Economic Behavior* **62**(2), 573–590 (2008)
- [17] Miguel, E., Gugerty, M.K.: Ethnic diversity, social sanctions, and public goods in kenya. *Journal of Public Economics* **89**(11-12), 2325–2368 (2005)
- [18] Nematzadeh, A., Ferrara, E., Flammini, A., Ahn, Y.Y.: Optimal network modularity for information diffusion. *Physical review letters* **113**(8), 088,701 (2014)
- [19] Newman, M.E.: Models of the small world. *Journal of Statistical Physics* **101**(3-4), 819–841 (2000)
- [20] Newman, M.E.: Spread of epidemic disease on networks. *Physical review E* **66**(1), 016,128 (2002)
- [21] Pastor-Satorras, R., Vespignani, A.: Epidemic spreading in scale-free networks. *Physical review letters* **86**(14), 3200 (2001)
- [22] Robinson, A.L.: Nationalism and ethnic-based trust evidence from an african border region. *Comparative Political Studies* Advanced Online Publication. doi: 0010414016628269 (2016)
- [23] Siegel, D.A.: Social networks and collective action. *American Journal of Political Science* **53**(1), 122–138 (2009)
- [24] Valente, T.W.: Social network thresholds in the diffusion of innovations. *Social networks* **18**(1), 69–89 (1996)
- [25] Valente, T.W., Davis, R.L.: Accelerating the diffusion of innovations using opinion leaders. *The Annals of the American Academy of Political and Social Science* **566**(1), 55–67 (1999)
- [26] Young, H.P.: Innovation diffusion in heterogeneous populations: Contagion, social influence, and social learning. *The American economic review* **99**(5), 1899–1924 (2009)

A Novel Approach to Predict Retweets and Replies Based on Privacy and Complexity-Aware Feature Planes

Kamini Garg, Valerio Arnaboldi and Silvia Giordano

Abstract An efficient tweet dissemination predictor for retweets and replies is central both to a better understanding of influentials (people and messages), as well as of how social media revenue models can be better monetized. Traditionally research concentrated on retweets popularity and information cascades while neglecting the importance of features richness and classification. We propose a novel approach that introduces feature planes for better prediction of single step tweet dissemination. We show that our model can achieve a quasi-perfect prediction. This promises to be a seminal step towards a better understanding of data dissemination in social networks.

1 Introduction

The widespread use of social networking sites like Twitter and Facebook allow users to generate and share information anywhere and anytime. The receiver of a message in such large scale networks has an option either to relay or forward it to his/her followers. In Twitter, this process is called retweeting and typically users retweet a message if they consider it interesting and worth sharing with others. A sequence of retweets along the network is called information cascade. Due to this process of sharing, a large amount of content is generated in Twitter and opened the door for new research directions in the field of information spreading, advertising, recommendations and social data mining. For example, online advertisers could use this information for efficient targeted marketing campaigns. Media companies could learn how to effectively generate buzz for new films or shows. Political groups could

Kamini Garg (e-mail: kamini.garg@supsi.ch)✉ · Silvia Giordano (e-mail: silvia.giordano@supsi.ch)

University of Applied Sciences and Arts of Southern Switzerland, Manno and University of Lugano, Switzerland

Valerio Arnaboldi (e-mail: valerio.arnaboldi@iit.cnr.it)
IIT-CNR, Pisa, Italy

learn who they should try to influence in order to spread their message as far as possible. Further, event results can also be predicted with good approximation.

Existing works in this area mainly tried to predict complete cascades by utilizing specific aspects of information diffusion like social network structure, temporal properties, profile features and topical features [6, 13, 14, 16] but none of them successfully combined all these features together and, more importantly, they do not quantify the importance of different features for retweet prediction. We argue that a fundamental knowledge of different feature planes (defined as a group of features with similar cost in terms of privacy and complexity to acquire), their individual and combined contribution in retweet prediction has to be analyzed first for better prediction of information diffusion. Therefore, we take a step back and identify feature planes based on their complexity to acquire and privacy intrusiveness and study their impact on retweet prediction to build a better understanding of diffusion. We believe that a deeper understanding of single step diffusion can be utilized as important building blocks for future models to precisely predict the complete information cascades. Our approach allows a very effective single step retweet prediction and quantifies the influence of different feature planes on prediction results. Further, as opposed to other works, we also take into account reply for information propagation by predicting the likelihood to reply to a particular tweet. To the best of our knowledge, our work is the first one that deeply studies the importance and impact of different planes of features on retweet and reply prediction. We summarize our contributions as follows:

- We define different planes of features that differ in complexity to acquire and level of privacy required.
- We introduce a novel approach that predicts the likelihood of tweet, retweet, and reply for a given tweet and user by using different feature planes. As opposed to other works, our approach does not limit the retweet and reply prediction to tweets generated by friends of a target user but predicts the likelihood for a generic tweet.
- We provide a deep understanding of single step information diffusion in social networks.
- Our results show high precision for both retweet and reply for different planes and also present that user twitter activities feature plane provides the highest precision. Further, our results are also seminal to researchers by providing the trade-off between high prediction accuracy and privacy.

In next sections, we first give an overview of the state of the art (Section 2) and describe the dataset used in Section 3. Further, we introduce our feature planes classification (Section 4) and then our approach to classify retweet and reply in Section 5. In Section 6, we validate our model and present the results for different feature planes. Finally, we conclude the paper along with future directions in Section 7.

2 Related Works

Although some initial work has been done to model complete diffusion cascades in social media [6, 13], researchers have recently argued that cascades might be inherently unpredictable, due to the high number of factors, either internal or external to the network [12], that affect the outcome of diffusion [11, 15]. For this reason, predicting the exact pattern of diffusion of a piece of information starting from a given node in the network remains challenging.

Most of the works in literature are mainly focusing on the analysis of specific aspects of information diffusion in social networks, such as whether diffusion will grow in future or not [4], the impact of content sentiment on diffusion [5], and the effect of features related to items or users on content popularity [8, 16]. In this paper, instead of trying to predict complete information cascades, we decided to firstly estimate whether a given user will retweet or reply a single tweet. Although some work has been already done in this research direction, the proposed solutions are still rather incomplete in terms of a number of features utilized and analysis of their impact on the diffusion process. For example, the work by Yuan and colleagues [17] is focused on the impact of social relationships and tie strength on the probability of diffusion. The work aims at sorting the friends of a user by their likelihood to retweet or reply its tweets and, does not specifically address information diffusion. Pezzoni et al. [14] analyzed the impact of temporal features and popularity indicators on the diffusion. The results indicate that content age and its visibility in the homepage of the user strongly influence the probability of resharing. Yet, compared to our work, this approach is particularly focused on temporal variables and does not consider other feature aspects.

Another research area related to the analysis of single step diffusion is from the perspective of personalized tweet recommendation. This approach aims to recommend tweets that could be interesting to the users instead of predicting whether users will reshare them in the future. On this line of research, Chen et al. use several features related to users profiles and their similarity, the content of tweets, and the social relationships between users to recommend existing tweets to users [3]. A similar solution is also proposed by Hong et al. [9]. Differently from these approaches, we focus on both retweet and reply prediction and we study the impact of different feature planes by considering the complexity to download each feature plane and the level of privacy it requires. Further, as opposed to existing works, our proposed model does not limit the prediction of retweet and reply for tweets that are generated by the friends of the user rather predicts it for any generic tweet.

3 Data Set Description

We used data collected from the Twitter activity of a large sample of about 2M users. The dataset was downloaded by Arnaboldi et al. in 2013 [1]. The dataset has been crawled through Twitter REST API, starting from a popular user in the network and then downloading all the available information about user's tweets and profile. Subsequently, the crawler iteratively downloaded same information for all

the followers and friends of the user who downloading phase was terminated. The obtained dataset is a large snowball sample of the network, with some degree of randomization due to the parallelization of downloads and the choice to start from a user with a large number of followers. This makes this dataset particularly suitable for the analysis of social interactions and information dissemination within groups of connected users, especially for those within small groups, for which the crawler possibly downloaded their complete network of social contacts.

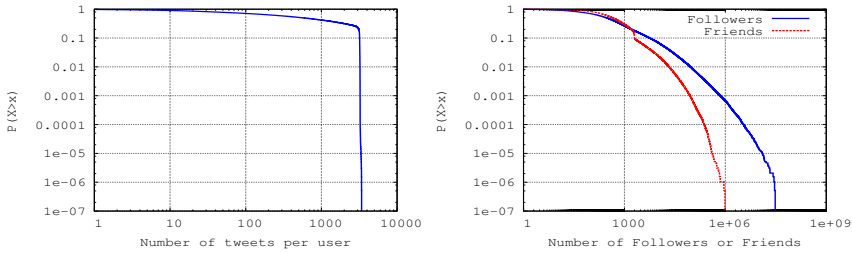
For each user in the dataset, we have the complete history of tweets and retweets they posted on Twitter up to the limit of 3,200 tweets per user imposed by Twitter REST API. In total, the dataset contains more than 2 billion tweets, each of which is characterized by creation time, the id of the creator, textual content, the number of retweets it received, information about geo-location, and the set of entities it contains such as hashtags, ids of other users mentioned in the text, URLs, etc.. In addition, each tweet also contains information about possible directed interactions between users. For retweets, this includes the id of the user who created the original tweet (i.e., the tweet that has been retweeted) and the creation time of the original tweet. For replies, the tweet includes the id of the user who replied. The profile data downloaded for each user includes general user's information, such as user's name, description, geo-location, language, a personal URL, as well as some statistics about user's Twitter usage like a total number of tweets created, and the number of followers and friends.

Figure 1a depicts the CCDF of the number of tweets created by each user. It is worth noting that the distribution is truncated around 3,200¹ for the limit imposed by Twitter API. Nonetheless, the number of Twitter users who reach this limit are roughly 10% of the total number of users in the dataset. This means that for the majority of people we have the complete history of tweets they created. In addition, for the users who created more than 3,200 tweets, we have a large sample of their recent Tweeting history. Figure 1b depicts the CCDF of the number of followers and friends per user. Both graphs show a very long tail, with a very small fraction of users in the dataset reaching about one million of friends, and more than 20 million followers. This is a typical aspect of social networks and indicates the validity of our sample.

4 Feature Planes

To model a person's likelihood to retweet and reply, we propose different planes of features and extract them from Twitter data according to the increased complexity to acquire them and their privacy intrusiveness. From the privacy point of view, we consider how much information do we need to mine and reveal about a user in order to predict retweet and reply. The consideration of privacy during Twitter data mining

¹ For some users, the number of tweets is slightly larger than 3,200 since we performed multiple downloads during the set-up process of the crawler, which lasted roughly one month, and we might have obtained the additional tweets generated during this month for some users.



(a) CCDF as a function of the number of tweets per user (b) CCDF as a function of the number of followers and friends per user.

Fig. 1: Complementary cumulative distribution function of the number of tweets created and number of friends and followers per user

is also highlighted in recent studies [10] [7]. Based on these contexts, we propose different feature planes starting from profile features to sentiment analysis of tweets.

Figure 2 presents different planes of features considered in the paper starting from *Profile* to *Global* plane. Please note that in each feature plane, we also consider features associated to the current Tweet. With Tweet features, we intend to examine the popularity of the original tweet and time sensitivity [13]. Other Tweet features we considered in each plane are the sentiment of the tweet, the number of embedded mentions and URLs obtained through tweet inspection.

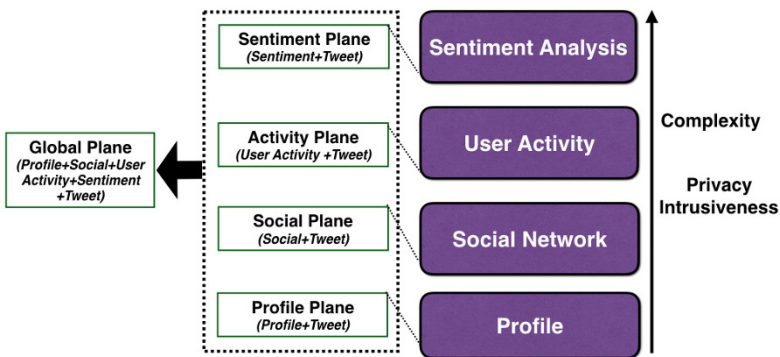


Fig. 2: Feature planes based on the complexity to acquire and privacy intrusiveness starting from user profile features to sentiment analysis of tweets.

4.1 Profile Plane

Features associated with this plane are the easiest to acquire using public Twitter API². From the Twitter profile of a user, we intend to get information about the user's account history like the length of user screen name, availability of URL, user description and image on his/her profile. We hypothesize that users with longer account history and rich profile information may be more active on Twitter, therefore, it is more likely to predict their likelihood to retweet and reply. Additionally, we also capture social information of the user from their profile by extracting the number of friends, the number of followers and the number of groups a user is associated with (listed count). Finally, from user profile we also consider the activity of users through their status counts (how many tweets users has published recently) and favorite counts (how many tweets has been marked favorite by a user) features.

4.2 Social Plane

Features in this plane represent the social ties of a person. Intuitively, if a person has more friends and followers then he/she has a higher probability to retweet and reply. Recent works also show that potential of retweeting as an act of friendship and to gain followers [2]. In this context, we process each user's network of friends and followers and extract features related to the number of friends, the number of followers, ratio of a number of friends to the number of followers and, ratio of a number of non-friends to the number of followers. As compared to *Profile* plane features, *Social* plane features are difficult to acquire and more privacy intrusive as we look into the entire social network of users.

4.3 Activity Plane

This plane captures all past and recent activities of Twitter users to predict their willingness to retweet and reply. We assume that if a person exhibits more activity on Twitter, then it is more likely that he/she will retweet and reply. We also quantify user's activity with respect to their friends, followers, and strangers like descriptive statistics for tweets per follower, friends, and strangers. *Activity* plane features are even more difficult to acquire and more privacy intrusive because we inspect all tweets of users to extract statistics about their past tweet, retweet and reply behavior with other users. In this plane, we capture both past and recent activities of users. For past activities, we utilize all available tweets up to current time while for recent activities we only take into account past month data (i.e. four weeks).

² The profile data of a user can be accessed through a single Twitter API call.

4.4 Sentiment Plane

The features associated with this plane are the most computational costly and privacy intrusive as compared to other planes because, in this case, we inspect the content of each tweet and process them to find associated positive, negative or neutral sentiment. Similar to *Activity* plane features, we also extract all past and recent sentiments of tweets and also quantify tweet sentiments for friends, followers, and strangers. To measure the overall sentiment of a set of tweets (or retweets/replies) in a day, we define sentiment index SI in Equation 1 where s^+ represents positive sentiment and s^- presents negative sentiment values in a day. To calculate SI , we first detect all English tweets from data set for each user and perform sentiment analysis on day-wise tweets using TextBlob³. To calculate SI values, we only consider tweets whose sentiments can be classified through TextBlob library. Likewise, we calculate SI values for each day of the tweets corresponding to each user.

$$SI = \frac{\sum s^+ - \sum s^-}{\sum s^+ + \sum s^-} \quad (1)$$

4.5 Global Plane

This plane combines all features from *Profile*, *Social*, *Activity*, and *Sentiment* planes along with Tweet features. With the help of this plane, we intend to study the aggregated impact of all feature planes on retweet and reply prediction.

5 Multi-Classification Prediction Model

5.1 Pre-processing and Training Data Generation

From the collected dataset, we only consider English tweets and also annotate each tweet as a tweet, retweet or reply and call them type 0, 1 and 2 respectively. In this way, we create ground truth to check the accuracy obtained from our prediction results. Each tweet signifies no diffusion while retweet and reply represent the single-step diffusion. From processed data set with English tweets, we calculate features over time to capture possible changes in retweet and reply behaviors with time and generate a time series for each variable to be more precise in predictions. Further, we aggregate these features in a weekly time window for *Activity* and *Sentiment* plane and store them in an SQLite database separately. The weekly aggregation was a good trade-off between precision and complexity because with the daily aggregation the complexity of the model was too high for the amount of data that we have.

Utilizing our database, we create a final set of features for a given user and tweet pair $\langle u, tw \rangle$ to train our prediction model. To create *Activity* and *Sentiment* plane features for $\langle u, tw \rangle$ pair, we extract data only till the current time of the tweet tw . Please note that, since the features for *Profile* and *Social* planes do not change with

³ textblob.readthedocs.io/en/dev/quickstart.html

time for a given user, they remain static for a given $\langle u, tw \rangle$ pair. Table 1 presents the format of feature sets for all planes given as input to train our model.

Table 1: Feature Set Input For Prediction Model

Feature Plane	Feature Set
Profile Plane	$\langle UserID, TweetFeatures, ProfileFeatures, TweetType \rangle$
Social Plane	$\langle UserID, TweetFeatures, SocialFeatures, TweetType \rangle$
Activity Plane	$\langle UserID, TweetFeatures, UserActivityFeatures, TweetType \rangle$
Sentiment Plane	$\langle UserID, TweetFeatures, SentimentFeatures, TweetType \rangle$
Global Plane	$\langle UserID, TweetFeatures, ProfileFeatures, SocialFeatures, UserActivityFeatures, SentimentFeatures, TweetType \rangle$

5.2 Prediction Model

We tested a number of classification algorithms such as Logistic regression, Random Forests models and Support Vector Machines, and chose regularized gradient boosting XGBoost to classify tweet, retweet, and reply. We chose XGBoost as it showed more stable performance across target variables and it does not require feature space specification, therefore, not affected by feature selection performance.

Our implementation of Gradient Boosting Method is based on the Python library XGBoost⁴. To classify tweet, retweet, and reply, we utilize multi-class classification using the *softmax* objective function. Further, we tried a set of parameter combinations to prevent overfitting using three parameters, *eta* that determines the learning rate, *gamma* regulating the sensitiveness to training examples, and the *number of rounds*. Based on different experiments, we set *eta* and *gamma* as 0.1 and 0 respectively. We apply *10-fold cross-validation* to select an appropriate *number of rounds* based on the multi-classification error rate. For a given $\langle u, tw \rangle$ pair, our model predicts the likelihood of diffusion by classifying tweet, retweet, and reply. If our model predicts retweet and reply for a $\langle u, tw \rangle$ pair then, the single-step diffusion will occur otherwise, there will be no diffusion due to the likelihood of tweet predicted by our model for the given pair.

⁴ xgboost.readthedocs.io/en/latest/python/python_intro.html

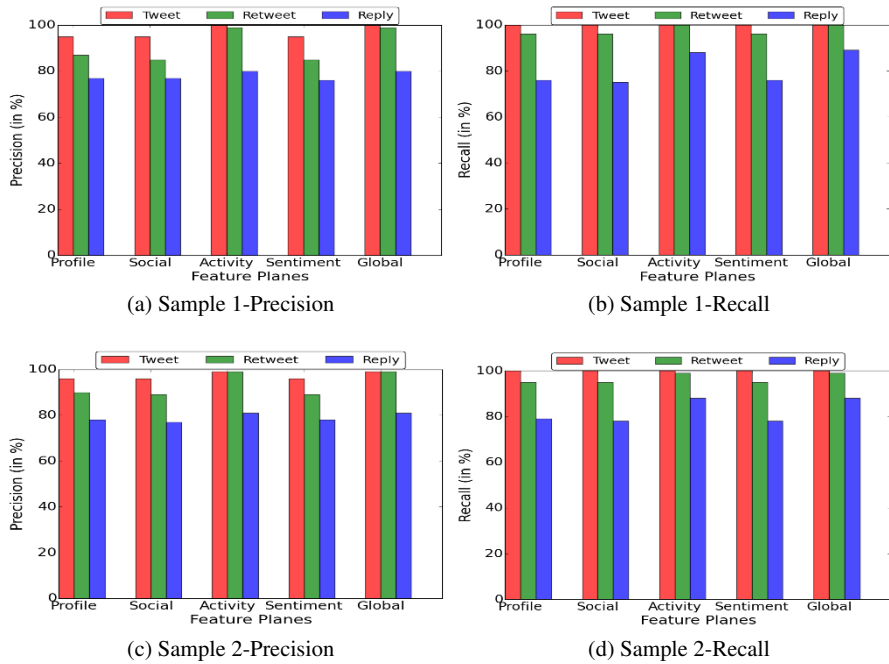


Fig. 3: Precision and Recall obtained from different models utilizing different planes of features starting from *Profile* to *Global* plane.

6 Results and Discussion

We measure precision and recall obtained from XGBoost model for the tweet, retweet, and reply classification. We split the set of tweets into a training and a testing set based on the timestamp of the tweets. The training set consists 60% of all tweets and the remaining 40% of the data is used to evaluate the prediction quality. We tested our model on two different samples (Sample 1 and Sample 2) of dataset selected based on different time intervals with 673,858 and 1,031,116 tweets respectively. Sample 1 data consists only one-month tweets of users while Sample 2 have all tweets of users for all years. For each sample, we tested model accuracy for different planes of features starting from *Profile* plane to *Global* plane. Figure 3 presents the precision and recall obtained from both samples for all feature planes for the tweet, retweet, and reply classification. From our results, we observe that for both samples, *Activity*, and *Global* plane features outperform and provide retweet, and reply classification with 99% and 82% precision and 99% and 80% recall values. Further, our model is also able to correctly classify tweets with high precision (99%) and recall (100%) values. These results show that if we process and mine more information about users, the model becomes more precise in classifying tweet, retweet, and reply. We also observe that our model performs slightly better (2%) in *Profile* plane as compared to *Social* and *Sentiment* planes. This happens because, in *Profile* plane, we have more information about the user in terms of the number of status messages, association to



Fig. 4: Confusion Matrix for Tweet, Retweet, and Reply classification obtained from our model utilizing different feature planes starting from *Profile* to *Global* for Sample 1.

groups while *Social* plane only has high-level information about friends and followers and *Sentiment* plane only considers sentiment of tweets. From above results, we observe the importance of the profiles of users and their activities on Twitter.

The precision and recall obtained using *Activity* and *Global* planes are equivalent and show that the maximum precision can be achieved only by considering user activities on Twitter i.e. *Activity* plane features. The inclusion of other feature planes such as *Profile*, *Social*, and *Sentiment* do not further improve prediction results. Our results highlight that only with *Profile* plane features, we can still achieve good accuracy thus, our model also takes away the complexity of large data processing and privacy concerning issues. Finally, we also present the confusion matrix for both sample 1 and 2 in Figure 4 and 5 respectively. From both confusion matrix, we observe that our model is able to correctly classify tweets, retweets, and reply for all planes thus, confirms the results obtained from Figure 3.

Table 2 presents the most important features associated with each plane utilized by our prediction model. From Table 2, we observe that Tweet features are one of the most important features across all planes. The tweet related features that contribute the most to precise prediction results are the time of the tweet, a number of times the tweet has been retweeted (Retweet count) and length & sentiment of the tweet. Since Tweet features are associated with each plane, therefore, we also quantify their impact on model accuracy and observe that they contribute 30% to the overall model accuracy across all planes. Our prediction model obtains similar results for both samples (sample 1 one month data while sample 2 with years of data) across all planes. Therefore, our results show that only with one-month of the Twitter

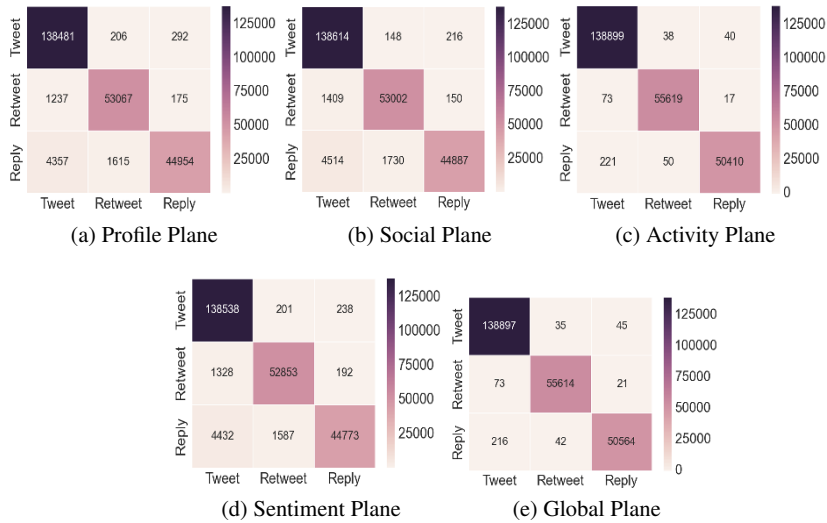


Fig. 5: Confusion Matrix for Tweet, Retweet, and Reply classification obtained from our model utilizing different feature planes starting from *Profile* to *Global* for Sample 2.

activity for a set of users is enough for accurate predictions. This result provides the implications for the amount of data required for the tweet, retweet, and reply classification and could be utilized in future diffusion models.

Compared to other resharing prediction models in the literature, we obtain sensibly higher accuracy values. For example, the model presented in [9], which is, to the best of our knowledge, the only model that can be directly compared to ours, obtains prediction accuracy lower than 80%. It is also worth noting that this model limits the prediction to tweets only generated by friends of the target users, whereas in our model we calculate the likelihood to retweet or reply a generic tweet, not necessarily generated by someone connected to the selected user.

Finally, we also validate the applicability of our prediction model for different time periods. To do this, we further group our testing data in the order of time (hour, day and week) after the last tweet of training data. For example, in the case of one hour, we only classify tweets that have been generated at max one hour after the last tweet in training data. Similarly, for days and week, we only classify those tweets that have been generated till the current day or week. From our results, we observe that for testing tweets generated up to one day after the last tweet of training data, our model classifies tweets, retweet, and reply with slightly higher precision (2%) for all planes except *Activity* and *Global* planes. In the case of *Activity* and *Global* planes, the precision obtained from our model was same across different time periods thus, show the preciseness and applicability of our model for different time periods and makes our model time independent. This happens mainly due to our rich dataset and

Table 2: Most Important Features For Different Planes

Profile Plane	Tweet time, # Followers, Tweet length, Twitter account age, # status messages, # Friends, Retweet count, Tweet sentiment, Listed Count, Length of user description
Social Plane	Tweet time, Ratio of friends and followers, # Friends, Tweet length, # Followers, Retweet count, Tweet sentiment
Activity Plane	Retweet count, Tweet length, Time elapsed since last Retweet, Tweet time, # Mentions, Tweet sentiment, STD of inter Retweet time, STD of # urls in Retweet, # Hashtags, Min. of inter Reply time, Mean of inter Tweet time, Time elapsed since last Tweet, Max. of total Retweets per follower, # Url
Sentiment Plane	Tweet time, Tweet length, Retweet count, Tweet sentiment, STD Retweet SI per follower, STD of Retweet SI, Max. of Retweet SI, Max. Retweet SI per follower, STD of Reply SI, STD of Tweet S, Entropy of Retweet SI, STD of Tweet SI per week, Entropy of Retweet SI
Global Plane	Retweet count, Tweet length, Time elapsed since last Retweet, Tweet time, # Mentions, Tweet sentiment, STD of inter Retweet time, STD of # urls in Retweet, # Hashtags, Min. of inter Reply time, Mean of inter Tweet time, Time elapsed since last Tweet, Max. of total Retweets per follower, # Url

consideration of both recent and overall past activities of Twitter users and the right features selected from our Gradient Boosting model.

7 Conclusions and Future Work

In this paper, we present a novel approach to predict the likelihood to tweet, retweet, and reply based on different feature planes. Our approach provides the deeper understanding of the diffusion process and quantifies the impact of different feature planes: *Profile*, *Social*, *Activity*, *Sentiment*, and *Global*. We propose feature planes based on the complexity to acquire and privacy intrusiveness. Differently from existing solutions, our model enables tweet, retweet and reply prediction for any generic tweet and does not limit the prediction for tweets generated by someone connected to the user. We validated our model on two different samples of the large-scale Twitter dataset and observe that our model outperform existing works for all planes by providing higher precision and recall for both samples. From our results, we also observe that *Activity* and *Global* feature planes outperform as compared to other feature planes. Further, our results are also seminal to researchers by providing the trade-off between high prediction accuracy and privacy.

In future, we plan to utilize user-level and tweet-level interest similarities for retweet and reply prediction by creating knowledge graph of topics from tweets. Finally, we intend to use our model to predict complete cascades in Twitter.

Acknowledgements This work was supported by the EU FP7 ERANET program under grant CHIST-ERA-2012 MACACO.

References

- [1] Arnaboldi, V., Conti, M., Passarella, A., Pezzoni, F.: Ego networks in twitter: an experimental analysis. In: Proceedings of IEEE INFOCOM, pp. 3459–3464 (2013)
- [2] Boyd, D., Golder, S., Lotan, G.: Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In: System Sciences (HICSS), 2010 43rd Hawaii International Conference on, pp. 1–10. IEEE (2010)
- [3] Chen, K., Chen, T., Zheng, G., Jin Ou and Yao, E., Yu, Y.: Collaborative personalized tweet recommendation. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, pp. 661–670 (2012)
- [4] Cheng, J.C., Adamic, L., Dow, A.P., Kleinberg, H.M., Leskovec, J.: Can cascades be predicted? In: Proceedings of the 23rd international conference on World wide web, pp. 925–936 (2014)
- [5] Ferrara, E., Yang, Z.: Quantifying the effect of sentiment on information diffusion in social media. In: ArXiv preprints (2015)
- [6] Galuba, W., Aberer, K., Chakraborty, D., Despotovic, Z., Kellerer, W.: Outtweeting the twitters - predicting information cascades in microblogs. In: Proceedings of the 3rd Wconference on Online social networks, pp. 3–3 (2010)
- [7] Gan, D., Jenkins, L.R.: Social networking privacywhos stalking you? *Future Internet* **7**(1), 67–93 (2015)
- [8] Hoang, T.A., Lim, E.P.: Virality and susceptibility in information diffusions. In: Sixth International AAAI Conference on Weblogs and Social Media (2012)
- [9] Hong, L., Doumith, A., Davison, B.D.: Personalized retweet prediction in twitter. In: 4th Workshop on Information in Networks (2012)
- [10] Kelley, P.G., Cranshaw, J.: Conducting research on twitter: A call for guidelines and metrics (2013)
- [11] Martin, T., Hofman, J.M., Sharma, A., Anderson, A., Watts, D.J.: Exploring limits to prediction in complex social systems. In: Proceedings of the 25th International Conference on World Wide Web, pp. 683–694 (2016)
- [12] Myers, S.A., Zhu, C., Leskovec, J.: Information diffusion and external influence in networks. In: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 33–41 (2012)
- [13] Petrovic, S., Osborne, M., Lavrenko, V.: Rt to win! predicting message propagation in twitter. In: ICWSM (2011)
- [14] Pezzoni, F., An, J., Passarella, A., Crowcroft, J., Conti, M.: Why do i retweet it? an information propagation model for microblogs. In: Proceedings of the 5th International Conference on Social Informatics, pp. 360–369 (2013)
- [15] Salganik, M.J., Dodds, P.S., Watts, D.J.: Experimental study of inequality and unpredictability in an artificial cultural market. *Science* **311**(5762), 854–856 (2006)
- [16] Yang, J., Counts, S.: Predicting the speed, scale, and range of information diffusion in twitter. In: In 4th International AAAI Conference on Weblogs and Social Media (2010)
- [17] Yuan, N.J., Zhong, Y., Zhang, F., Xie, X., Lin, C.Y., Rui, Y.: Who will reply to/retweet this tweet?: The dynamics of intimacy from online social interactions. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, pp. 3–12 (2016)

Least Squares Method for Diffusion Source Localization in Complex Networks

Mohammed Lalou and Hamamache Kheddouci

Abstract Studying diffusion process in complex networks has become an important issue nowadays. This issue has been addressed for different objectives, including quickly detecting the diffusion outbreak, blocking the propagation, and localizing the diffusion source. In this paper, we are mainly interested in developing an efficient algorithm to estimate both the source and the start time of the diffusion, under the constraint that only a subset of nodes can be observed. In doing so, we use the *Ordinary Least Squares* method (*OLS*) on the data gathered at observers, taking advantage of the linear correlation between the relative infection time of a node and its effective distance from the source (Brockman [2]). The proposed algorithm ensures an estimation at few hops from the actual source. We show its efficiency through numerical simulations on both synthetic and real networks.

1 Introduction

Epidemics propagation in populations, virus cascading in computer networks, rumors spreading in social networks are considered as examples of diffusion process in complex networks. Studying this process has attracted much attention in recent years. This issue has been considered for different objectives, including: inferring the underlying diffusion network [3], maximizing the spread of influence [4], blocking the contagion diffusion [5], and locating the diffusion sources [11][8][14]. The last one has recently received much attention.

Mohammed Lalou e-mail: mohammed.lalou@gmail.com✉
Computer Sciences Department, University of Bejaia, Algeria

Mohammed Lalou
Institute of Sciences and Technology, University Center of Mila

Mohammed Lalou · Hamamache Kheddouci(e-mail: hamamache.kheddouci@univ-lyon1.fr)
LIRIS, UMR5205 CNRS, University of Claude Bernard Lyon1- Lyon, France

Localizing the source of diffusion is an important task that has many applications in several areas, such as: identifying the culprit by the authorities [11], determining the patient-zero of a pandemic [8], disclosing the person who started a rumor (in a social network) [13], finding the administrator of a cyber-attack [14], etc.

An intuitive solution to efficiently localize the diffusion source is to observe the state of all nodes in the network. This assumption has been considered in a first class of works, where the state of all nodes is supposed known by giving a snapshot of the diffusion spread [8, 14]. With this hypothesis, authors in [14] proposed a Maximum-Likelihood Estimator (MLE) for trees and extended for general graphs. The estimator depends on a defined metric denoted rumor centrality. As well, in [8], a probabilistic algorithm has been developed using the dynamic message-passing equations. However, Observing all nodes fails for two reasons: first, it is generally unfeasible as the most of the networks of interest are very large, and second, it is not cost-efficiently as controlling nodes has usually a cost. This constraint has been considered in the second generation of works, which tries to localize the source under the constraint that the state of only a subset of nodes can be observed [10, 11, 13, 16]. Under this constraint, a MLE that maximizes the localization probability has been developed in [11]. This estimator is optimal for trees and suboptimal for general graphs. In [10], the diffusion source has been proven to be the Jordan center of the tree formed by the set of observed nodes. In [9], a two-stage algorithm has been proposed, which first identifies the most likely candidate cluster to contain the source, and then tries to locate the source within this cluster using a MLE.

In this paper, we consider the source localization problem under the constraint that only a subset of nodes can be observed. Thus, two main questions arise: the first one concerns the design of an efficient observation model, and hence we ask about which nodes we should observe in order to efficiently control any diffusion outbreak. This question has been considered in [11, 13, 16], and it has been shown that the strategy for selecting nodes to be observed affects the source estimation performance through a comparison between different strategies. The second question involves the development of an efficient method to accurately localize the diffusion source using only the partial information gathered at the observed nodes. Our answers to these questions provide our main contributions summarized as follow:

1. Proposing an efficient parameterized observation model taking advantage of the network structural properties, which is an efficient way to have a good observation given that the network is the conduit for the diffusion. For this purpose, we use the *critical nodes* of the network [6]. The model provides a useful tool which deals with different diffusion objectives together, namely, detecting, blocking and localizing the source of diffusion.
2. Developing an efficient approach to estimate both the diffusion source and the time zero of the diffusion (if it is unknown). For this purpose, we first use the proposed observation model to gather the infection time information. Then, taking advantage of a fundamental diffusion property showed by Brockman [2], where the infection time of a node and its effective distance from the source are linear correlated, we use a linear regression method, namely the *Minimum Least Squares* method to estimate both the source node and the time zero. We note that

the proposed method is model-free and works for any observation model. Also, it does not depend on the underlying diffusion model.

3. Conducting simulations on both synthetic and real-world networks to show the efficiency of our approach in localizing the diffusion source .

The rest of the paper is organized as follows. In *Section 2*, we give the definitions used in the paper. *Section 3* describes our proposed observation model for sparse networks. In *Section 4*, we detail our approach for localizing the diffusion source, and then we evaluate its performance in *Section 5*. Finally, we close up the paper by some perspectives and future works.

2 Definitions and notations

In this section, we introduce the needed concepts for locating the diffusion source.

The Diffusion model in networks: The network on which diffusion occurs is modelled as an undirected graph $G = (V, E)$, where V is the set of nodes, and E is the set of edges (if G is unknown, we can infer it using inferring algorithms [3]). To model the diffusion in the network, we adopt the well-known *Susceptible-Infected (SI)* model, where each node is in state: (i) *Susceptible (S)*: the node is able to be infected, or (ii) *Infected (I)*: the node can spread the contagion further. Once a node is infected, it will stay infected forever. This model covers most of the possible situations. In addition, a propagation probability p_{uv} is associated with each edge uv , and the diffusion process is the following. At time t , each infected node u tries to infect all its neighbors. Each neighbor node v becomes infected with a probability p_{uv} , and will remain infected throughout. The process continues until there is no possible propagation. We assume that the diffusion outbreak occurs by a single node, called the source node $s \in V$, at time t_0 . This diffusion model is general enough to accommodate various scenarios encountered in practice.

Critical Nodes: The critical nodes of a graph are those whose deletion significantly degrades the graph connectivity according to a predefined metric [6]. Here we are interested in the so-called *Component Cardinality Constraint Critical Node Problem*, defined as follows. Given a graph G and an integer L , find the minimum set of nodes whose deletion disconnects G on connected component of at most L nodes.

Effective distance: Given two adjacent nodes u and v in a graph $G = (V, E)$, the effective distance d_{uv} between u and v is defined as follows:

$$d_{uv} = 1 - \log_2(p_{uv})$$

where p_{uv} is the propagation probability from u to v . This measure implies that a small fraction p_{uv} is effectively equivalent to a large distance between u and v , and vice versa. Based of this concept, we define the path-effective distance of a path $P(u_1, u_n) = \{u_1, u_2, \dots, u_n\}$ as:

$$\lambda[P(u_1, u_n)] = \sum_{uv \in \Gamma} d_{uv}$$

the effective distance D_{uv} from an arbitrary node u to another node v is defined as:

$$D_{uv} = \min_{P(u,v)} \lambda[P(u,v)]$$

A diffusion process starting at node u is equivalent to a homogeneous wave propagated on the u -rooted tree formed by the shortest paths $P(u,v), \forall v \in V$ (see [2]).

3 Observation model for sparse graphs

Network controllability using only a subset of nodes, called *observers*, has been considered in [7]. The objective is to select a minimal set of nodes whose monitoring allows to control the network state and hence detect any diffusion. Using this idea, we propose to take as observers the critical nodes [6] of the network (the nodes whose deletion disconnects the network on components of at most L nodes). The motivation behind using these nodes as observers is that observing them:

1. ensures an early detection of any diffusion outbreak in the network, and hence makes easier the localization of the source since detecting the diffusion as soon as possible allows to minimize the number of infected nodes, and thus the nodes likely to be the source. This is possible since observing critical nodes ensures that each diffusion that spreads in the network is observed by at least one observer after at most L hops. Note that critical nodes are the connection between the components and any path between two component pass through them.
2. provides an easily locating of the part of the network where the outbreak occurred (the part containing the source), without the need for the direction of infection. For that, each observer, once observing the diffusion, it halts it and does not spread it further, and hence the diffusion will be contained only on the connected component bounded by the first critical nodes receiving the infection.
3. provides distributed and balanced control on the whole network, which we should ensure in the case where all nodes are equally likely to start the diffusion. Also, it allows different levels of network controllability with respect to the values of L (the size of component), which provides more flexibility and more potential.
4. it is cost efficient. In fact, observing nodes has usually a cost, which we aim to minimize, and in finding critical nodes we seek for the minimal set of nodes.

4 Diffusion Source localization

In this section, we detail our approach for locating the diffusion source. We assume that the diffusion process is initiated by a single node s at time t_0 . The time t_0 can be (i) known, this is the case, for example, of diffusions occurred due to disasters, where the disaster start time is known, or (ii) unknown, which is the general case. We also assume that any node in the network is equally likely to be the source a priori. Let $O = \{o_k\}_{k=1}^K \subset V$ be the set of K observer nodes. We denote the *active observers* $O_a \subseteq O$, the subset of observers that receives the infection, and C_a the infected

component *i.e.*, the part of the network where the diffusion is detected, and hence the part containing the source node. Based on the first time the observers become infected, C_a is easily identified as the component bounded by O_a since besides observing the diffusion, each observer is designated to stop and do not disseminate further the diffusion once observed. That allows the diffusion to be contained in only one component (denoted C_a). Accordingly, each observer $o_i \in O_a$ provides the time at which the infection is received. We denote $T_a = \{t_{o_k}\}_{k=1}^K$ the infection times of observers. In [2], it has been shown that the relative infection time of a node u is linear with its effective distance from the source s , so we have:

$$t_u = \alpha \cdot D_{su} + c \quad (1)$$

where t_u is the relative infection time of node u , D_{su} the effective distance from s to u . Based on this fundamental property, we estimate the real diffusion source using the well-known *Ordinary Least Squares method* [12].

4.1 Estimating the source node

In this section, we describe the use of the *OLS* method to locate the diffusion source. Note that given two random variables X (the independent variable) and Y (the dependent variable), and a set of n pairs of observations $\{Y_i, X_i\}$ where the value of Y and X are related by a linear equation: $Y = a + b \cdot X$, *OLS* estimates the parameters a and b of the "best fit" line to the observed data. The estimation is defined as the values which minimize the sum of the squared errors (for more details, see[12]). In our case, the independent variable X is the effective distance (D), and the dependent variable Y is the infection time (t). We recall that each observer saves its infection time, while its effective distance from the node supposed to be the source is computed using the underlying network. We consider the two cases, whether the time t_0 is known or not.

4.1.1 Case 1. The start time t_0 is known

In this case, we suppose that t_0 is known. To estimate the source node, we investigate all suspects node in C_a (the infected component). For each node $u \in C_a$, we compute the effective distance between node u and all observers $o_i \in O_a$. Thus, we have K pairs of observations $\{t_{o_i}, d_{uo_i}\}$, corresponding to the relative infection times and the effective distances between observers and node u . Then, we apply *OLS* on $\{t_{o_i}, d_{uo_i}\}_{i=1}^K$ to compute the parameter α as shown in line (6) of *Algorithm 0*, while compelling the line to pass through the point $\{t_0, 0\}$ ($c = t_0$). The parameter α is then used to compute the sum of the squared errors. We do the same with all suspect nodes, and the real source node is the node which minimizes the residual sums.

4.1.2 Case 2. The start time t_0 is unknown

Now, we consider the case where t_0 is unknown. Thus, we seek for both locating the diffusion source and estimating the initial time of the diffusion. Identifying the time zero has many advantages. For instance, it helps in discovering the real reasons of the diffusion by locating where the source was. Indeed, learning about the environment where the source was can lead to a good control of the diffusion. This is the case of epidemics, where determining exactly where the patient zero travelled and who they came into contact with, helps the epidemiologists to discover the origin of the infectious disease, to track its spread, and undertake procedures to isolate it. Note that harmful viruses often exist in some nidus¹, and the infection starts when the virus comes into contact with patient zero.

We proceed as for Algorithm 0, we investigate all nodes u in C_a and obtain the set of data $\{t_{o_i}, d_{u o_i}\}_{i=1}^K$ corresponding to the relative infection times of observers and their effective distances from u . Then, we compute the correlation coefficient between the effective distances and the infection times of observers. As the infection time of a node is linear with its effective distance from the diffusion source, then the most likely node to be the real source is the suspect node with the best value of the correlation coefficient (computed as shown in line (11) of Algorithm 0). In our case, the closer the coefficient is to 1, the better is the correlation, since the infection time increases as the effective distance increases ($0 < \rho \leq 1$). Also, the estimated start time \hat{t}_0 is the intersection between the regression line and the time-axis *i.e.*, $\hat{t}_0 = c$, since $t_u = c + \alpha \cdot D_{su}$ and $D_{su} = 0$ when $u \equiv s$.

5 Experimental results

In this section, we present simulation results using both synthetic and real-world networks to evaluate the performance of the proposed estimator for diffusion source localization. To model the virus spreading in the network, we adopt a discrete time *Susceptible-Infected* model (*SI*). The time is slotted, *i.e.*, divided into discrete slots, at time $t = 0$, there is only one infected node, called the source. A susceptible node $u \in C_a$ adjacent to any infected node v becomes infected with probability $p_{uv} \in (0, 1)$, at the beginning of the next time slot.

For synthetic networks, we consider the two well-known models, namely small-world and scale-free networks [15]. We first identify the observers (which are the critical nodes) using the heuristic described in [1], and then we run the estimator described in Section 4.1. Without loss of generality, in diffusion simulation we consider only the infected component with a predefined set of observers. All reported results are averaged over 100 independent runs. For each run, and since there is no prior knowledge of the source of diffusion, we randomly select a node to be the source. The main metric we use to evaluate the estimation accuracy is the *distance*

¹ Nidus is the long-term host -natural reservoir- of a pathogen of an infectious disease, such as animals like rats.

Algorithm 14 Diffusion Source Localization- t_0 is known

-
- 1: **Input:** a graph $G = (V, E)$ with propagation probability p_{uv} for each edge $uv \in E$, a set of k active observers $O_a = \{o_1, \dots, o_k\}$ and their infection times $T_a = \{t_{o_1}, \dots, t_{o_k}\}$, and the diffusion start time t_0 .
 - 2: **Output:** the estimated diffusion source s^* .
 - 3: $s^* \leftarrow \{\}, \tau \leftarrow +\infty$.
 - 4: **for** each node $u \in C_a$ **do**
 - 5: For each observer $o_i \in O_a$, compute the effective distances $D_{uo_i} = 1 - \log(p_{uo_i})$, and lets $D = \{D_{uo_1}, \dots, D_{uo_k}\}$.
 - 6: Compute the equation of the regression line for the independent variable D and the dependent variable T_a while forcing the line to pass through $\{t_0, 0\}$, as follows:

$$c = t_0, \quad \text{and} \quad \alpha = \frac{\sum (t_{o_i} - \bar{T}_a)(D_{uo_i} - \bar{D})}{\sum (D_{uo_i} - \bar{D})^2}$$

- 7: Let $\sigma = \sum_i [t_{o_i} - (c + \alpha D_{uo_i})]^2$ be the residuals sum returned by the line.
 - 8: **if** $\sigma < \tau$ **then**
 - 9: $s^* \leftarrow u$, and $\tau \leftarrow \sigma$.
 - 10: **end if**
 - 11: **end for**
 - 12: Return the estimated diffusion source s^* .
-

$error^2$, denoted θ . Different algorithms are implemented using C++, R (using *igraph* package) and Python (using *Networkx* package).

As our approach is based on the linear relationship between the relative infection time of a node and its effective distance from the source [2], we first show through experimentation the concreteness of this property. *Fig. 1.* clearly shows a strong linear correlation between the infection time of a node and its effective distance from the source on both small-world and scale-free networks of 1000 nodes.

When the start time is known, we select as estimated source the suspect node with the smallest residual sum, and when the start time is unknown, we select the node with the greatest correlation coefficient value. In order to evaluate this idea, we investigate, in *Fig. 2.(left)*, the influence of the distance from the source on both the correlation coefficient and the residual sum, considering a small-world network of 300 nodes and diameter of 22 hops, and where 20% of nodes are observed. Here the distance from the source is given by the ratio of the number of hops from the source and the network diameter. We can see that when the distance from the source increases, the correlation coefficient decays (it reaches 0.1 for a distance of 46% away from the source), while the residual sum increases (up to a max value 300 for a distance of 58% away from the source). In *Fig. 2.(right)*, we show the relationship between the localization accuracy and the number of observed nodes. We perform the simulation on 100 small-world networks of 300 nodes, and we take the percentage

² The number of hops between the actual source and the estimated source

Algorithm 15 Diffusion Source Localization- t_0 is unknown

- 1: **Input:** a graph $G = (V, E)$ with propagation probability p_{uv} for each edge $uv \in E$, a set of k active observers $O_a = \{o_1, \dots, o_k\}$ and their infection times $T_a = \{t_{o_1}, \dots, t_{o_k}\}$.
- 2: **Output:** the estimated diffusion source s^* , and the estimated start time of diffusion t_0 .
- 3: $s^* \leftarrow \{\}$, $\rho \leftarrow -\infty$.
- 4: **for** each node $u \in C_a$ **do**
- 5: For each observer $o_i \in O_a$, compute the effective distances $D_{uo_i} = 1 - \log(p_{uo_i})$, and lets $D = \{D_{uo_1}, \dots, D_{uo_k}\}$.
- 6: Compute the correlation coefficient between D and T_a as follows:

$$\rho^* = \frac{Cov(D, T_a)}{\sigma_D \sigma_{T_a}} = \frac{\sum_{i=1}^K (D_{uo_i} - \bar{D})(t_{o_i} - \bar{T}_a)}{\sum_{i=1}^K (D_{uo_i} - \bar{D})^2}$$

- 7: **if** $\rho^* < \rho$ **then**
- 8: $s^* \leftarrow u$, and $\rho \leftarrow \rho^*$.
- 9: **end if**
- 10: **end for**
- 11: Compute the regression line of the variables D and T_a according to the found source node s^* as follows:

$$\hat{t}_0 = \bar{T}_a - \alpha \cdot \bar{D}, \text{ where } \alpha = \frac{\sum (t_{o_i} - \bar{T}_a)(D_{uo_i} - \bar{D})}{\sum (D_{uo_i} - \bar{D})^2},$$

- 12: Return the estimated diffusion source s^* , and the estimated start time \hat{t}_0 .

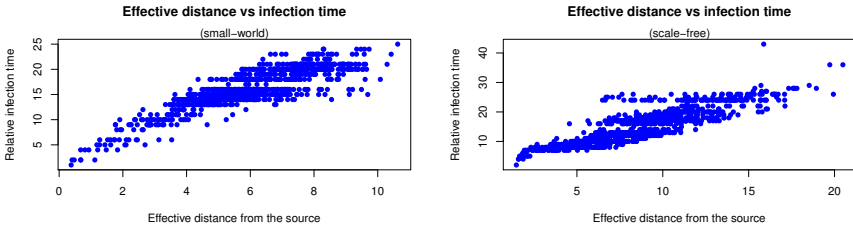


Fig. 1: Correlation between the relative infection time of a node and its effective distance from the source D for a small-world network (left), and a scale-free network (right) of 1000 nodes.

of solutions where the distance error $\theta \leq 1$. Clearly, we can see that the average error distance decreases when the number of observers increases (87% of solutions with $\theta \leq 1$ are reached when 58% of nodes are observed), which is explained by the fact that the more observations we make (*i.e.*, more observers we have), the more information we have about the diffusion, and hence the best is the regression line.

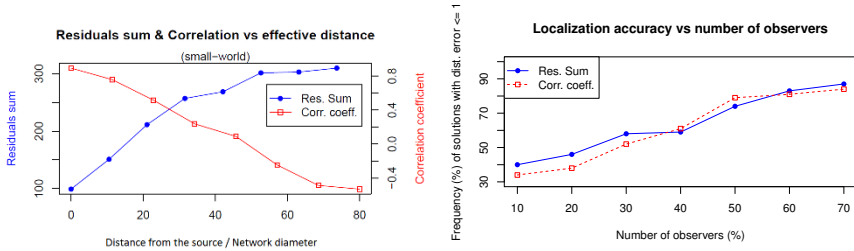


Fig. 2: (left) The influence of the effective distance on the residual sum and the correlation coefficient (right) The influence of the observer number on the locating accuracy in a small-world network on the case when t_0 is known (using residual sums) or not (using correlation coefficients).

5.1 The diffusion start time is known

Fig. 3. shows the accuracy of the proposed estimator through a histogram of 100 networks of 500 nodes, where 20% have been observed. The network diameter is between 14-32 hops for the small-world network, and 15-23 hops for scale-free. In more than 82% of runs, the estimator localizes the source with a distance error at most 2 hops. For scale-free networks, 75% of runs localizes the source in at most 3 hops from the actual source. Thus, a good performance has been noted for both small-world and scale-free networks. Note that some bins (of distance error 10 and 11) lie far away from the distance error values center. This is due to outliers.

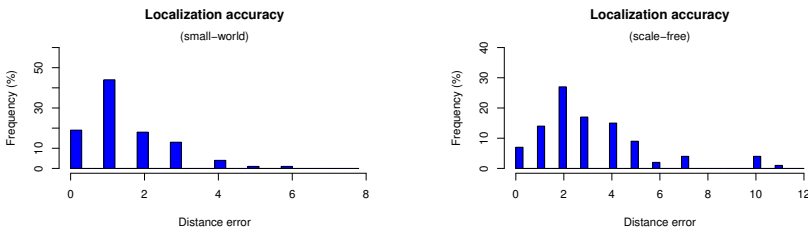


Fig. 3: A histogram of the source estimator accuracy for small-world and (left) and scale-free (right) networks of 500 when the start time is unknown.

5.2 The diffusion start time is unknown

Now, consider the case when t_0 is unknown. In this case, the estimated source is the node with the greatest correlation coefficient. Fig. 5. shows a histogram of distance error for 100 small-world and scale-free networks, of diameter average, respectively,

23 hops and 19 hops. The network size is 500 nodes. We can see that the method ensures a localization within at most 3 hops from the actual source with probability 95% for the small-world model while observing only 20% of nodes. For scale-free, 82% of runs ensure an estimation with a distance error at most 4 hops.

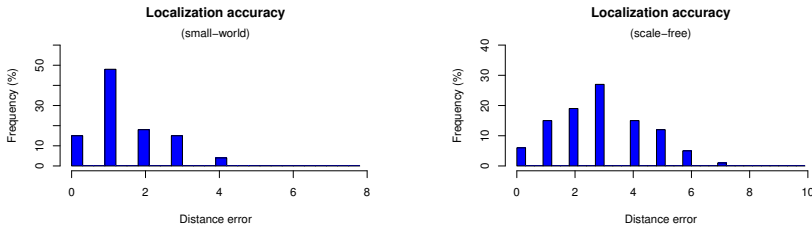


Fig. 4: A histogram of the source estimator error for 100 networks of small-world (left) and scale-free (right) networks of 500 nodes when t_0 is unknown.

Once the source localized, we can estimate the diffusion start time using the regression line. Histograms in Fig. 5. show that the estimated start time is only 15% away from the initial time with a probability of 84% and 73%, for small-world and scale-free networks, respectively. We note that we measure the time estimation using *time ratio*, which is the ratio of the time error and the total diffusion time. The time error is the number of time units between the estimated start time and the actual t_0 .

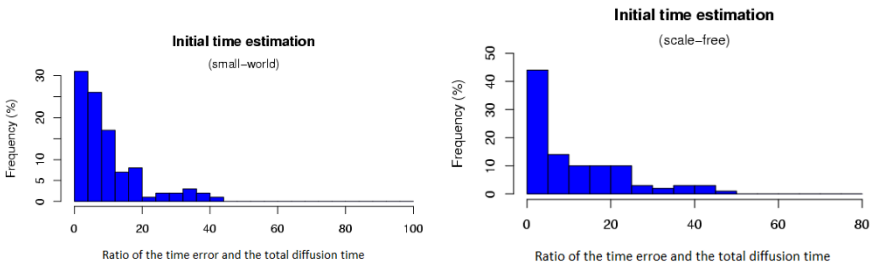


Fig. 5: Diffusion start-time estimation for small-world and scale-free networks of 500 nodes.

5.3 Real-world networks

In this section, we perform experimentation on real-world networks. Table 1 summarizes some properties of the infected component considered for two networks³.

³ The used benchmark can be download from <http://snap.stanford.edu/data/index.html> # email

Table 1: Different properties of the networks used in the experiments.

Network	n (# nodes)	m (# edges)	Diameter
Facebook net.	4 039	88 236	8
Email-Enron net.	10 500	109 488	10

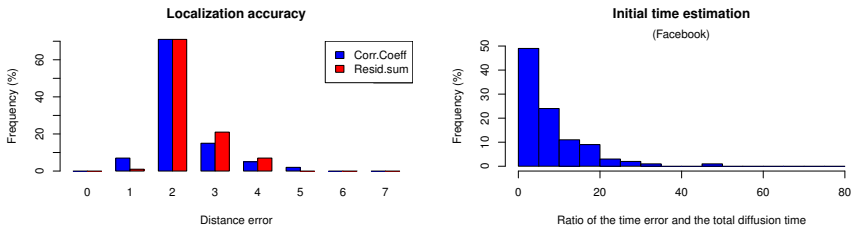


Fig. 6: (left) Source estimation accuracy when t_0 is known (Residual sums) or not (Correlation coefficients) of Facebook network (right) The diffusion start time estimation probability.

On Facebook where 10% of nodes are observed. Fig. 6. shows a localization accuracy of a distance at most 3 hops from the actual source in more than 90% of runs for both cases where t_0 is known or not. Also in more than 75% of runs, the start time estimation is less than 10% away from the initial time. On Email-Enron network (Fig. 7.), the source is located at a distance at most 3 hops in more than 70% of runs when the time t_0 is known, and more than 80% of runs when t_0 otherwise. The start time is estimated at less than 10% away from the real t_0 in 90% of runs.

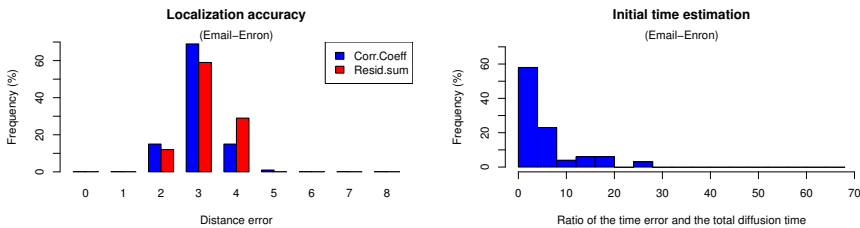


Fig. 7: (left) Source estimation accuracy when t_0 is known or not of Email-Enron network (right) The diffusion start time estimation probability.

6 Conclusion and Future works

The results in this work clearly show that using the linear regression analysis provides an efficient approach for locating the diffusion source when only partial observations are available. In fact, the proposed algorithm achieves a good record on estimating both the source and the start time of the diffusion. In order to demonstrate the effectiveness of our estimator, we have to compare it with existing estimators, especially the Maximum Likelihood Estimator [11]. Also, we have to enhance the observation model (using other metrics) to take into account networks which do not contain so many critical nodes. In the other, and as these encouraging results have been obtained using the most basic linear regression analysis approach, then more promising results can be expected using more sophisticated approaches where multiple parameters can be considered (the *OLS* method exploits only one parameter, namely the relationship between distance and infection time). Also, the *OLS* method has an important drawback, which is the sensibility to outliers (extreme observations) as observed in some diagrams. Thus, the use of more advanced approach such as robust regression methods helps in dealing with this impairment.

Acknowledgements This work is supported by Thomson Reuters in the framework of the Partner University Fund project : Cybersecurity Collaboratory: Cyberspace Threat Identification, Analysis and Proactive Response”. The Partner University Fund is a program of the French Embassy in the United States and the FACE Foundation and is supported by American donors and the French government.

References

- [1] Arulselvan, A., Commander, C.W., Elefteriadou, L., Pardalos, P.M.: Detecting critical nodes in sparse graphs. *Computers & Operations Research* **36**(7), 2193–2200 (2009)
- [2] Brockmann, D., Helbing, D.: The hidden geometry of complex, network-driven contagion phenomena. *Science* **342**(6164), 1337–1342 (2013)
- [3] Gomez Rodriguez, M., Leskovec, J., Krause, A.: Inferring networks of diffusion and influence. In: *Proceedings of the 16th ACM SIGKDD*, pp. 1019–1028. ACM (2010)
- [4] Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: *Proceedings of the 9th ACM SIGKDD*, pp. 137–146. ACM (2003)
- [5] Kuhlman, C.J., Tuli, G., Swarup, S., Marathe, M.V., Ravi, S.: Blocking simple and complex contagion by edge removal. In: *ICDM 2013*, pp. 399–408. IEEE (2013)
- [6] Lalou, M., Tahraoui, M., Kheddouci, H.: Component-cardinality-constrained critical node problem in graphs. *Discrete Applied Mathematics* **210**, 150–163 (2016)
- [7] Liu, Y.Y., Slotine, J.J., Barabási, A.L.: Controllability of complex networks. *Nature* **473**(7346), 167–173 (2011)
- [8] Lokhov, A.Y., Mézard, M., Ohta, H., Zdeborová, L.: Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Physical Review E* **90**(1), 012,801 (2014)
- [9] Louni, A., Subbalakshmi, K.: A two-stage algorithm to estimate the source of information diffusion in social media networks. In: *Computer Communications Workshops (INFOCOM WKSHPS)*, 2014 IEEE Conference on, pp. 329–333. IEEE (2014)
- [10] Luo, W., Tay, W.P., Leng, M.: How to identify an infection source with limited observations. *Selected Topics in Signal Processing*, IEEE Journal of **8**(4), 586–597 (2014)
- [11] Pinto, P.C., Thiran, P., Vetterli, M.: Locating the source of diffusion in large-scale networks. *Physical review letters* **109**(6), 068,702 (2012)

- [12] Rao, C.R., Toutenburg, H.: *Linear models*. Springer (1995)
- [13] Seo, E., Mohapatra, P., Abdelzaher, T.: Identifying rumors and their sources in social networks. In: *SPIE defense, security, and sensing*, pp. 83,891I–83,891I (2012)
- [14] Shah, D., Zaman, T.: Detecting sources of computer viruses in networks: theory and experiment. In: *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, pp. 203–214. ACM (2010)
- [15] Wang, X.F., Chen, G.: Complex networks: small-world, scale-free and beyond. *IEEE circuits and systems magazine* **3**(1), 6–20 (2003)
- [16] Zejnilovic, S., Gomes, J., Sinopoli, B.: Network observability and localization of the source of diffusion based on a subset of nodes. In: *Communication, Control, and Computing (Allerton), 2013 51st Annual Allerton Conference on*, pp. 847–852. IEEE (2013)

The effects of local network structure on disease spread in coupled networks

W. Vermeer, B. Head and U. Wilensky

Abstract Epidemiology has long used human interaction patterns to understand spreading dynamics. Recently network scientists have embraced the notion that these patterns are best described using a complex multi-layered system, a network of networks, yielding a stream of literature focused on understanding spreading in such coupled systems. Adding this macro level perspective to disease spreading, focusing on the interaction among systems, has shifted focus away from the role of local (within-system) structure. In this paper, using a multi-level Agent-based model, we highlight the importance of the local structure in determining spreading dynamics in coupled settings. We show that the local dynamics in both the focal and neighboring networks, play a significant role in determining focal dynamics. As both are driven by the local structure this highlights a need for incorporating structural details across all levels for accurate modeling of disease spreading dynamics.

1 Introduction

Understanding the spread of disease in populations has long been a focus of the field of epidemics. The inherent difficulty of measuring disease spread has resulted in a tendency to rely on modeling to gain insight into epidemics. Traditional epidemic models assumed a compartmentalization of the population into different states

W.H. Vermeer (e-mail: wouter.vermeer@northwestern.edu)✉

Department of Psychiatry and Behavioral Science, Northwestern Institute on Complex Systems and Department of Learning Sciences, Northwestern University, Chicago, IL

B.Head (e-mail: bryan.head@u.northwestern.edu)

Department of EECS, Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL

U. Wilensky (e-mail: uri@northwestern.edu)

Department of EECS and Department of Learning Sciences, Northwestern Institute on Complex Systems,
Northwestern University, Evanston, IL

(Susceptible, Infected, Removed) and assumed homogenous mixing of such compartments. A vast body of work created since has incorporated a network perspective in modeling of epidemics (eg. [17, 18]). The underlying assumption in these studies is that the network structure serves as the infrastructure for propagation and therefore bounds the dynamics that can occur. Adoption of such a network perspective has yielded an increased understanding of disease spreading behavior.

The notion that spreading phenomena are based on more complex interaction patterns has more recently gained traction in network science. Resulting in studies of cascades in inter-dependent [5, 6], multi-layered [2, 4, 7, 14], and multiplex networks [11]. Specifically in the field of physics, considerable progress has been made in modeling and in understanding how coupling between networks affects the dynamics in multi-layered systems [4, 12]. This body of work has highlighted that the inter-layer connections –both in terms of structure [10] and strength [8, 11]– strongly impacts the spreading dynamics [7], highlighting the importance of adopting a coupled system perspective for spreading phenomena.

While previous examples are all part of the set of coupled system studies, capturing the idea that spread occurs in systems which consist of multiple coupled systems, the way in which the system is described varies strongly across studies. For example, a multiplex network setting assumes a single set of nodes (agents or actors) connected by multiple types of ties, whereas multi-layered and inter-dependent settings assume two (or more) systems, each with a set of separate nodes and ties that are (partially) connected by an inter-system layer.

Especially in social contexts, which are based on the behavior of people, the multi-layered perspective seems to naturally fit. People have a variety of drivers for multiple types of interactions, and mobility patterns (and thus interaction patterns) that are strongly bound by geographical constraints. It is easy to interact with those that are geographically proximate, e.g. within a city of residence. Although long geographical jumps are possible (for example by air travel) such jumps are often much less likely. Therefore, the human interaction system is both fundamentally multiplex (many types of interactions) and multi-layered (mobility on different scales). In this system, locally dense networks across the globe are coupled by means of occasional long jumps. The inherent structure of this system makes any propagation process based on the human interaction a prime example of a phenomena that should be studied using a coupled networks approach.

In line with this reasoning, [2] is a prime example of adopting a coupled system approach in epidemiology, and the model presented is a big step forward from the single system model. It should be noted that, albeit being multi-layered, this is not a model of coupled networks as the local layer consists of a gravity model rather than a network model. While this might have been a modeling choice, as network data with this granularity is hard to obtain, it is indicative of a general issue that applies to most coupled network research. As the scope shifts from a single networked system towards a system of coupled networks, the focus shifts from characteristics of the single network towards the characteristics of layer that connects the networks; from the local structure towards the structure of inter-system layer. In doing so the lessons learned from the local structure seem to be more and more forgotten and/or ignored.

There are many studies that have shown that, in single network settings, the network structure is a critical factor if one wants to understand, predict, and steer spreading dynamics. For example, it is known that shorter average path lengths greatly increase spreading potential [21], skewed degree distributions allow for even faster and more widespread disease cascades [1, 18] and that local clustering improves local spreading but hampers widespread disease cascades [19]. Yet in coupled network studies these local influences are commonly oversimplified, receive little attention, and are by no means systematically addressed. This raises the question whether, in the context of coupled networks, the local structure indeed plays no role (as suggested by [15]), or whether this role is falsely being ignored.

2 Methodology

Exploring the role of local network structure on disease spread in a coupled setting requires a model consisting of two main components; a system consisting of coupled network structures, and a disease spreading mechanism. We incorporate these two elements in an agent-based model (ABM) in NetLogo [22], and using LevelSpace [13] we adopt a multi-level modeling approach [16] for the coupled network scenarios.

2.1 *The structure of the system*

Building on the notions put forward by [2]) we create a system that consists of two types of layers: the “within-city” layer and the “between-city” layer. The within-city layer describes the structure of a single city which consists of a population of 1000 individuals which are connected in a fixed network structure. The network structure is one of the classical network topologies; Erdős-Rényi [9], scale-free [3], small-world [21] with a rewire probability of 0.05, or a regular ring lattice. The between-city layer consists of a model that captures the effects of coupling, each within-city layer is modeled separately and is connected by means of the between-city layer. Therefore the between-city layer acts as a bridge between the within-city models, effectively making this a multilevel model.

In this study we are interested in the effects of the local structure, the structure of the within-city layers, on disease spread dynamics in coupled settings. We know from previous literature that the inter-system (between-city) structure and strength are critical factors that influence the local dynamics, therefore we aim to reduce the impact of this layer as much as possible. We do so by simplifying the between-city model in three ways. First, we assume that there are only two coupled cities. Second, we assume that any between-city interaction will occur randomly. Both assumptions reduce the complexity of the between layer structure, of which a schematic representation can be found in Figure 1. Third, we assume that the spreading dynamics within and between cities are the same. More details on the dynamics can be found in the next section. Note that the third assumption implicates that the type of ties within and between cities are the same. Therefore one could model this as single

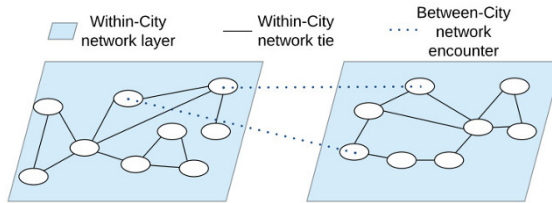


Fig. 1: In our model two within-city networks ($n = 1000$) with a fixed structure are coupled by randomly occurring encounters across the layers

giant network, where every individual in one cluster (city) is connected to every individual in the other cluster (be it with lower weights). Note that the resulting model would have orders of magnitude more links than the multi-level approach adopted in our study. For our parameter-set, in which cities are relatively small, the number of links in a single network would increase from 10,000 (5000 links in each city), to 1,010,000. This growth in the number of links would significantly increase the computational resources required, indicating that a multilevel modeling approach is far more powerful and scalable in coupled network settings.

2.2 description of disease spreading rules

In line with traditional compartment models, we assume individuals can be in one of four states: **S**usceptible, **E**xposed, **I**nfected, or **R**emoved (SEIR). All individuals are by default in the susceptible state. At the beginning of the simulation, two individuals in the focal city are exposed to the disease, effectively seeding the disease to 0.2% of that city. By interacting with susceptible and infected alters, individuals can then move from Susceptible \rightarrow Exposed \rightarrow Infected \rightarrow Removed states.

We assume that disease spread is caused by interactions (encounters) rather than the network structure itself. One can imagine the network structure as describing the structure of friendships, this structure provides the infrastructure of interactions. This means that having a friend that is sick does not put one directly at risk, however, interacting with that friend does. It is therefore the encounters in the network which drive the spread of disease, not the structure itself. We assume that during each time-step (tick) of the simulation, each Exposed and Infected individual has a certain number of encounters with its network neighbors. The number of such encounters is drawn from a Poisson distribution with a mean that is conditional on the state of the actor which can be varied in our model. Exposed individuals have a mean encounter rate of c_{ES} while infected individuals have a mean encounter rate of c_{IS} . We assume that the social activity (number of encounters) of individuals depends on how sick they are, hence Exposed (asymptomatic) individuals will have a higher number of encounters than Infected (symptomatic) ones.

The neighbors encountered are chosen randomly and independently; a neighbor

may be encountered multiple times in a single tick. Note that this means that the number of encounters an individual has is completely independent of their degree. This ensures that varying degree does not directly influence the rate at which the disease spreads. When comparing different network structures, keeping the encounter rate independent of degree ensures that any differences we observe are a result of the different network structures rather than different distributions of encounter rates. An example to illustrate: if encounter rates were proportional to degree, almost all individuals in the scale-free network would have a very low encounter rate (due to their low degree) while all individuals in the ring network would have the same, mid-sized encounter rate. This would make it impossible to distinguish if the observed effects are caused by variations in network structure or encounter rate.

When an Exposed (or Infected) individual encounters susceptible neighbors they become exposed with a given probability, which depends on the state of the individual that encountered them (whether the source is exposed (i_{ES}) or infected (i_{IS})). Exposed individuals automatically become infected after a certain duration which is drawn from an exponential distribution with mean $1/\delta$, and infected individuals become removed after a certain duration also drawn from an exponential distribution with mean $1/\delta$.

In line with [19] all experiments use the following parameters:

- mean degree (for all network types): 10
- c_{ES} – mean number of encounters for exposed: 4
- c_{IS} – mean number of encounters for infected: 1.25
- i_{ES} – probability of infection from exposed: 0.05
- i_{IS} – probability of infection from infected: 0.06
- $1/\epsilon$ – mean duration of exposed: 15
- $1/\delta$ – mean duration of infected: 15

As stated prior, disease dynamics follow the same logic in both layers (between-city and within-city). Rather than adding ties and increasing the pool from which encounters are pulled, the between-city model will redirect a certain percentage of the within-city encounters to be with individuals in the neighboring city. The reasoning behind redirection rather than addition is that adding between-city encounters would effectively change the rate at which disease can spread, which would make comparison across scenarios invalid. In our simulations 1% of the within-city encounters are redirected to the other city, meaning that within-city encounters are reduced to 99% of their initial rate in the single non-coupled city scenarios.

Selection of between-city encounters occurs completely random and independently, where any individual in one city can encounter any individual in the other city. For the purpose of this paper this way of modeling the between-city network is most applicable, yet, future work should be performed that compares different methods of connecting cities in order to understand interaction effects between the local (within-city) and the inter-system (between-city) structures.

2.3 Differential equation model

To create a base-line of disease spreading behavior we compare the within-city Agent-based model (ABM) with the classic SEIR compartmental model based on differential equations (DE). Similar to the ABM, in the DE model the population is divided into four segments: susceptible (S), exposed (E), infected (I), and removed (R). Also similar to the ABM, the susceptible population becomes exposed at a rate based on the infection rate and encounter rate of the exposed and infected populations. The differential equations encoding these relationships are given in the following equations:

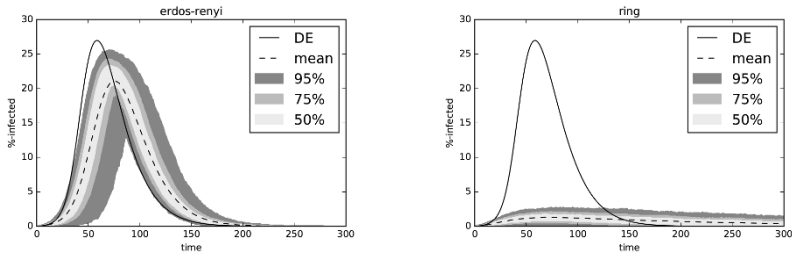
$$\begin{aligned}\frac{dS}{dt} &= -(c_{ES}i_{ES}E + c_{IS}i_{IS}I)S \\ \frac{dE}{dt} &= (c_{ES}i_{ES}E + c_{IS}i_{IS}I)S - \epsilon E \\ \frac{dI}{dt} &= \epsilon E - \delta I \\ \frac{dR}{dt} &= \delta I\end{aligned}$$

3 Results

To see if the simulation model behaves as intended, we start our analysis by reproducing the study conducted in [19] in a single network setting. We find that, in comparison, disease dynamics in our model (Figure 2) are stretched out over a longer period of time but follow corresponding trends across various structures. The observed delay is to be expected given our cities are 5x larger than those in the original work. This makes it more time consuming for the disease to reach saturation, which is indeed what we observe. As our disease spread dynamics are in line with [19], this serves as a sign that the agent-based simulation model is behaving as intended.

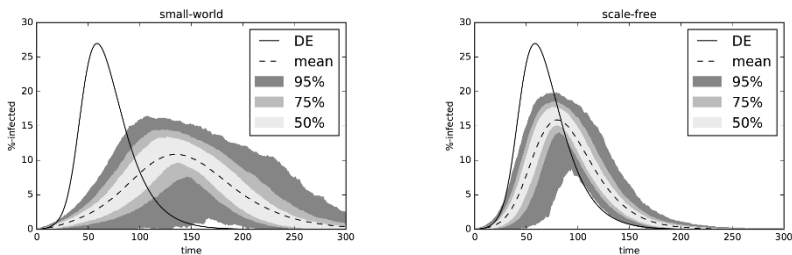
The single city results show that the spreading dynamics in the ABM differ significantly from those of a Differential Equation (DE) model; the peak load is much lower and occurs much later. Note that, even though the DE model effectively allows any individual to come into contact with any other individual, the number of encounters in the network model is fixed to be the same as in the DE. Therefore these differences do not stem from a reduced number of encounters in the network settings. Instead, the observed differences in spreading speed arise from localized connections and local clustering. The higher clustering increases the chance of inefficient encounters—from sick to sick—, reducing the effective spreading rate [19].

We continue the analysis by using the ABM to study the effects of coupling of within-city networks. While adding inter-city ties effectively adds a second mode of spreading (not only within but also between cities) we correct for the potential effects of such an increase in connectivity by keeping the rate at which individuals encounter others equal across all scenarios. The results (shown in Figure 3) reveal that the effect of coupling on the focal city dynamics is strongly conditional on the structure of the focal city. On the one hand, in cities with Scale-free and Erdős-Rényi networks, coupling does not result in any observable effect on disease spread



(a) Disease spreading dynamics in a single **Erdős-Rényi** network

(b) Disease spreading dynamics in a single **Ring** network



(c) Disease spreading dynamics in a single **Small-world** network

(d) Disease spreading dynamics in a single **Scale-free** network

Fig. 2: This figure shows the disease spreading dynamics in cities with varying within-city networks. The top (and bottom) percentiles are depicted in greyscale for a total of 1000 simulation runs in the Agent-based model. The dynamics of the differential equation of the same disease are plotted in blue.

dynamics. On the other, in cities with a small-world or ring networks, the spreading seems to be improved due to coupling. This is in line with previous work claiming coupled networks can suffer from increased volatility [20]. These results suggest that the effect of coupling on the focal city’s dynamics is strongly dependent on the within-city network structure of the focal city.

It is interesting to note that the focal cities affected by coupling are those that have structures with otherwise highly localized, and thus slow, spreading dynamics. This might suggest that random pathways facilitated by the between-city layer (individuals encountered in the neighboring city are chosen randomly) allow for long jumps which are otherwise unavailable in the focal network structure. This suggests that coupling effectively reduces the diameter of the focal city network via the between-city layer. A more intuitive explanation is that due to the slow spread within the focal network there is enough time for a second order spread—from the focal city to the neighboring city and back to the focal city—to occur before the within-city dynamics have saturated the focal city. The ring network (Figure 3b) clearly shows a second peak of spreading after the initial peak seems to flatten. This suggests the presence of the latter described second order spreading, in which the neighboring city causes

reseeding in the focal city.

These results indicate that the timing of epidemics across coupled networks seems to play a crucial role in the effects of such coupling. As the timing of an epidemic is directly related to where a disease starts, the seed becomes a critical aspect in our simulation. Seeding the focal city, as has been the case in previous analysis, causes the epidemic in the neighboring city to occur with a lag. This lagging reduces the potential impact of the neighboring city on the focal city and consequently the effects of coupling will likely be dominated by the epidemic dynamics within the focal city. To increase the potential effects of coupling we adjust our seeding location and repeat the previous analysis. Now, rather than seeding the focal city, the neighboring city will be seeded. The results (Figure 4) show that when the disease originates from the neighboring city the effects of coupling become much more apparent, resulting in a variety of dynamics in the focal city. When the focal city's epidemic is lagging behind those of the origin city—the city which was seeded with disease—the opportunities for secondary infections increase substantially, but the extent to which they occur depends on the disease growth rate in the the origin city. As we know this growth rate is determined by the local structure (see Figure 2) the observed variance in coupling effects should be attributed to the within-city structure in the origin city.

4 Discussion

Previous research has identified that both network structure and coupling of networks as drivers which can have significant effects on the local dynamics of disease spread. The focus on understanding the effects of coupling has shifted the attention away from the local structure as a driver, resulting in little systematic connection between these two bodies of work. Consequently, the effects of local network structure seem to be poorly integrated in the coupled network literature, both in terms of describing the structure of the local layers of interaction as well as the interaction of such local structures with the inter-layer structure [10]. While both could be addressed using the methodology presented in this paper, the scope of this paper is on highlighting the role of local structure in a coupled network setting.

By means of an Agent-Based Model of two coupled cities we have shown that local growth dynamics, caused by the local within-city structure, plays a crucial role in understanding if and how coupling will affect the focal disease spreading dynamics. While the relevance of the local (within-city) structure of the focal city has been identified in both single [17] as well as in coupled network settings [10], we find that the local (within-city) dynamics of the neighboring city also impacts the focal spreading dynamics. This indicates that simply knowing the focal city's structure and the way in which it is coupled to other cities is not sufficient for understanding spreading behavior. We find that the dynamics in neighboring cities, which depend on the neighboring city's local structure and the dynamics in the neighbor's neighbors, play a critical role in focal city's spreading dynamics. The feedback among cities not only indicates that the structural details in each of the local (within-city) layers matters, but also that dynamics of the focal city cannot be accurately considered without incorporating the coupled perspective.

Our results further emphasize the critical role of the effectiveness of the between-city layer. We find that a sufficient amount of time is needed for the coupling to become effective. This amount is conditional on both the focal growth rate (driven by within-city structure) and neighboring growth rate (driven by neighboring within-city structure). When the focal city's disease load is saturated it will not likely be affected by anything from the outside, making coupling a less important factor. This draws the attention to path dependence as a driver of spreading in coupled networks. If enough time is available, coupling can become efficient and has a strong effect on focal spreading dynamics. This observation is in line with previous work that identifies coupling strength as a key driver for coupling effects [8, 11].

While our model is conceptual in nature, there are interesting implications for health policy that can be devised from it. A comparison among seeding locations (the comparative plots are not included in this paper but can be done by comparing Figure 3 to Figure 4) indicates that for structures with relatively slow disease spreading (Small-world, Ring) a scenario that has a seed outside the focal city results in earlier and higher peak loads in the focal city, compared to the same scenario in which the focal city is seeded. Therefore, outside infections provide a higher risk for the focal population. In concrete terms, our results suggest that reducing disease load within a city (or country) is best achieved by preventing coupling, and this indeed seems to be a strategy implemented to prevent global pandemics like the 2014 Ebola

spread. However, as very small coupling probabilities have significant effects and complete decoupling seems infeasible, the effectiveness of such strategies will be limited, especially as global travel increases over time. When complete uncoupling is not an option it seems that reducing outbreaks in neighboring cities is more critical for controlling the dynamics in the focal city.

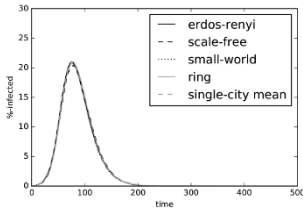
This is somewhat in conflict with the current way in which health policy is implemented; based on local agencies (be it the city, state, country) with local data and dynamics. Our results suggest a different approach with global coordination, in which the coupling of networks is considered and a global intervention strategy is implemented, not only because it is socially desired, but because it is in each local network's own self interest.

Acknowledgements Research reported in this publication was supported by the National Institute On Drug Abuse of the National Institutes of Health under Award Number P30DA027828, the National Science Foundation under Award Number NSF IIS-1441552, and the Northwestern Institute on Complex Systems. The content is solely the responsibility of the authors and does not necessarily represent the official views of any of the supporting agencies.

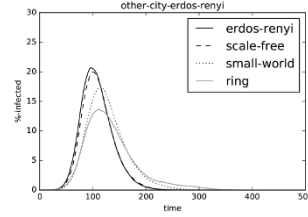
References

- [1] Albert, R., Jeong, H., Barabasi, A.L.: Error and attack tolerance of complex networks. *Nature* **406**(6794), 378–382 (2000)
- [2] Balcan, D., Colizza, V., Gonçalves, B., Hu, H., Ramasco, J.J., Vespignani, A.: Multiscale mobility networks and the spatial spreading of infectious diseases. *Proceedings of the National Academy of Sciences* **106**(51), 21,484–21,489 (2009)
- [3] Barabasi, L., Albert, R.: Emergence of scaling in random networks. *Science (New York, N.Y.)* **286**(5439), 509–512 (1999)
- [4] Boccaletti, S., Bianconi, G., Criado, R., del Genio, C.I., Gomez-Gardees, J., Romance, M., Sendia-Nadal, I., Wang, Z., Zanin, M.: The structure and dynamics of multilayer networks. *Physics Reports* **544**(1), 1–122 (2014)
- [5] Brummitt, C.D., DSouza, R.M., Leicht, E.A.: Suppressing cascades of load in interdependent networks. *Proceedings of the National Academy of Sciences* **109**(12), E680–E689 (2012)
- [6] Buldyrev, S.V., Parshani, R., Paul, G., Stanley, H.E., Havlin, S.: Catastrophic cascade of failures in interdependent networks. *Nature* **464**(7291), 1025–1028 (2010)
- [7] De Domenico, M., Granell, C., Porter, M.A., Arenas, A.: The physics of spreading processes in multilayer networks. *Nature Physics* (2016)
- [8] Dickison, M., Havlin, S., Stanley, H.E.: Epidemics on interconnected networks. *Physical Review E* **85**(6) (2012)
- [9] Erdős, P., Rényi, A.: On random graphs. *Publ. Math. Debrecen* **6**, 290–297 (1959)
- [10] Gao, J., Buldyrev, S.V., Stanley, H.E., Havlin, S.: Networks formed from interdependent networks. *Nature Physics* **8**(1), 40–48 (2011)
- [11] Gómez, S., Díaz-Guilera, A., Gómez-Gardeñes, J., Pérez-Vicente, C.J., Moreno, Y., Arenas, A.: Diffusion dynamics on multiplex networks. *Physical Review Letters* **110**(2) (2013)
- [12] Havlin, S., Stanley, H.E., Bashan, A., Gao, J., Kenett, D.Y.: Percolation of interdependent network of networks. *Chaos, Solitons & Fractals* **72**, 4–19 (2015)
- [13] Hjorth, A., Head, B., Wilensky, U.: "LevelSpace NetLogo extension". <http://ccl.northwestern.edu/levelspace/index.html>. Evanston, IL: Center for connected learning and computer based modeling, northwestern university. (2015)
- [14] Kivela, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. *Journal of Complex Networks* **2**(3), 203–271 (2014)

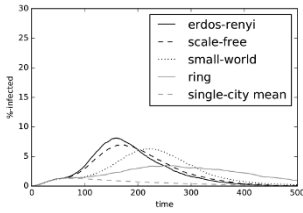
- [15] Mata, A.S., Ferreira, S.C., Pastor-Satorras, R.: Effects of local population structure in a reaction-diffusion model of a contact process on metapopulation networks. *Physical Review E* **88** (2013)
- [16] Morvan, G.: Multi-level agent-based modeling - a literature survey. arXiv:1205.0561 [cs] (2012)
- [17] Newman, M.E.J.: Spread of epidemic disease on networks. *Physical Review E* **66**(1) (2002)
- [18] Pastor-Satorras, R., Vespignani, A.: Epidemic spreading in scale-free networks. *Physical Review Letters* **86**(14), 3200–3203 (2001)
- [19] Rahmandad, H., Sterman, J.: Heterogeneity and network structure in the dynamics of diffusion: Comparing agent-based and differential equation models. *Management Science* **54**(5), 998–1014 (2008)
- [20] Vespignani, A.: Complex networks: The fragility of interdependency. *Nature* **464**(7291), 984–985 (2010)
- [21] Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. *Nature* **393**(6684), 440–442 (1998)
- [22] Wilensky, U.: Netlogo. <http://ccl.northwestern.edu/netlogo/>. center for connected learning and computer-based modeling, northwestern university, evanston, il. (1999)



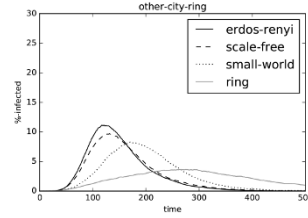
(a) Mean disease spread for a focal city with a **Erdős-Rényi** network, for various structures in the neighboring city



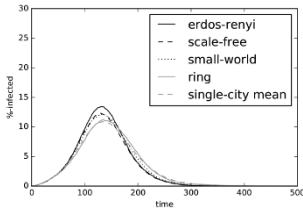
(a) Mean disease spread for a focal city with a **Erdős-Rényi** network, for various structures in the neighboring city



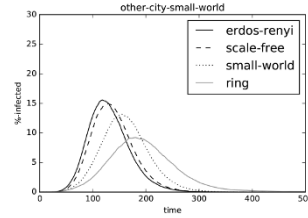
(b) Mean disease spread for a focal city with a **Ring** network, for various structures in the neighboring city



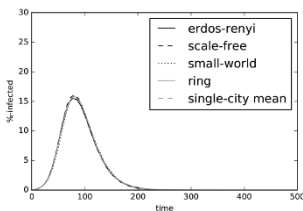
(b) Mean disease spread for a focal city with a **Ring** network, for various structures in the neighboring city



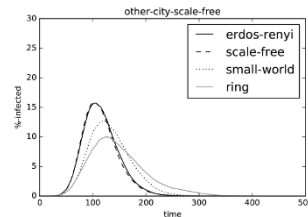
(c) Mean disease spread for a focal city with a **Small-world** network, for various structures in the neighboring city



(c) Mean disease spread for a focal city with a **Small-world** network, for various structures in the neighboring city



(d) Mean disease spread for a focal city with a **Scale-free** network, for various structures in the neighboring city



(d) Mean disease spread for a focal city with a **Scale-free** network, for various structures in the neighboring city

Fig. 3: This figure compares the disease spread dynamics in the focal city in scenarios in which the **focal** city is seeded with disease, while the network structure of the neighboring city is varied.

Fig. 4: This figure compares the disease spread dynamics in the focal city while the **neighboring** city is seeded with disease, while the network structure of the neighboring city is varied.

The Accuracy of Mean-Field Approximation for Susceptible-Infected-Susceptible Epidemic Spreading with Heterogeneous Infection Rates

Bo Qu and Huijuan Wang

Abstract The epidemic spreading over a network has been studied for years by applying the mean-field approach in both homogeneous case, where each node may get infected by an infected neighbor with the same rate, and heterogeneous case, where the infection rates between different pairs of nodes are also different. Researchers have discussed whether the mean-field approaches could accurately describe the epidemic spreading for the homogeneous cases but not for the heterogeneous cases. In this paper, we explore if and under what conditions the mean-field approach could perform well when the infection rates are heterogeneous. In particular, we employ the Susceptible-Infected-Susceptible (SIS) model and compare the average fraction of infected nodes in the metastable state, where the fraction of infected nodes remains stable for a long time, obtained by the continuous-time simulation and the mean-field approximation. We concentrate on an individual-based mean-field approximation called the N-intertwined Mean Field Approximation (NIMFA), which is an advanced approach considered the underlying network topology. Moreover, for the heterogeneity of the infection rates, we consider not only the independent and identically distributed (i.i.d.) infection rate but also the infection rate correlated with the degree of the two end nodes. We conclude that NIMFA is generally more accurate when the prevalence of the epidemic is higher. Given the same effective infection rate, NIMFA is less accurate when the variance of the i.i.d. infection rate or the correlation between the infection rate and the nodal degree leads to a lower prevalence. Moreover, given the same actual prevalence, NIMFA performs better in the cases: 1) when the variance of the i.i.d. infection rates is smaller (while the average is unchanged); 2) when the correlation between the infection rate and the nodal degree is positive. Our work suggests the conditions when the mean-field approach, in particular NIMFA, is more accurate in the approximation of the SIS epidemic with heterogeneous infection rates.

Bo Qu (e-mail: b.qu@tudelft.nl)✉ · Huijuan Wang (e-mail: h.wang@tudelft.nl)✉
Delft University of Technology, Mekelweg 4, 2628CD, Delft, The Netherlands

1 Introduction

By considering the system components as nodes and the interactions or relations in between nodes as links, networks have been used to describe the biological, social and communication systems. On such networks or complex systems, viral spreading models have been used to describe processes e.g. epidemic spreading and information propagation [8, 10, 13, 20]. The Susceptible-Infected-Susceptible (SIS) model is one of the most studied models. In the SIS model, each infected node infects each of its susceptible neighbors with an infection rate β . The infected node can be recovered with a recovery rate δ . Both processes are independent Poisson processes. The ratio $\tau \triangleq \beta/\delta$ is called effective infection rate, and when τ is larger than the epidemic threshold τ_c , the epidemic spreads out with a nonzero fraction of infected nodes in the metastable state. The average fraction of infected nodes y_∞ in the metastable state, ranging in $[0, 1]$, indicates how severe the influence of the virus is: the larger the fraction y_∞ is, the more severely the network is infected.

In this paper, we concentrate on deriving the average fraction y_∞ of infected nodes in the metastable state. Although the continuous-time Markov theory can be used to obtain the exact value of y_∞ , the number of states is too large to be solved in a large network [12]. Hence, the derivation of the average fraction y_∞ of infected nodes in the metastable state mostly relies on mean-field theoretical approaches. The first approach to study the SIS model in complex networks is a degree-based mean-field (DBMF) theory, also called heterogeneous mean-field (HMF) approximation, proposed by Pastor-Satorras et al. [14], which assumes that all nodes with the same degree are statistically equivalent, i.e. the infection probabilities of those nodes are the same. An individual-based mean-field (IBMF) approximations, called the N-Intertwined Mean-Field Approximation (NIMFA), of the SIS model is then introduced [19] with the only assumption that the state of neighboring nodes is statistically independent. NIMFA, taking the network topology into account, turns out to be more precise on different types of networks for the classic SIS model with the homogeneous infection rates[7] while comparing to the DBMF approximation. However, as discussed in [4, 15, 22], the infection rates could be heterogeneous, i.e. the infection rates between different pairs of nodes could also be different. The accuracy of NIMFA with heterogeneous infection rates has not yet been discussed.

In this paper, we explore the influence of the heterogeneous infection rates on the precision of NIMFA. In particular, we compare the average fraction y_∞ of infected nodes as a function of the effective infection rate τ computed by NIMFA to that obtained by the continuous-time simulations of the exact SIS model when the infection rates are heterogeneous but the recovery rate is the same for all nodes. In fact, the effective infection rate τ refers to the average infection rate divided by the recovery rate in the SIS model with heterogeneous infection rates. We set the average infection rate to 1 and tune the recovery rate δ to control the effective infection rate τ . We consider both the independent and identically distributed (i.i.d.) and the correlated heterogeneous infection rates in different network topologies.

2 Preliminary

2.1 The N-Intertwined Mean-Field Approximation

The N-Intertwined Mean-Field Approximation (NIMFA) is so far one of the most accurate approximations of the SIS model that takes into account the influence of the network topology. For the classic SIS model with the homogeneous infection rate β and recovery rate δ . The single governing equation for a node i in NIMFA is

$$\frac{dv_i(t)}{dt} = -\delta v_i(t) + \beta(1 - v_i(t)) \sum_{j=1}^N a_{ij} v_j(t) \quad (1)$$

where $v_i(t)$ is the infection probability of node i at time t , and $a_{ij} = 1$ or 0 denotes if there is a link or not between node i and node j . The governing equation (1) can be extended to the heterogeneous case:

$$\frac{dv_i(t)}{dt} = -\delta v_i(t) + (1 - v_i(t)) \sum_{j=1}^N \beta_{ij} a_{ij} v_j(t) \quad (2)$$

where $\beta_{ij} = \beta_{ji}$ is the infection rate between node i and j . In the steady state, defined by $\frac{dV(t)}{dt} = 0$ where $V(t) = [v_1(t) \ v_2(t) \ \cdots \ v_N(t)]^T$, $\lim_{t \rightarrow \infty} v_i(t) = v_{i\infty}$ and $\lim_{t \rightarrow \infty} V(t) = V_\infty$, we have

$$\left(\frac{1}{\delta} \text{diag}(1 - v_{i\infty})BA - I\right)V_\infty = 0 \quad (3)$$

where A is the $N \times N$ adjacency matrix with elements α_{ij} , I is the $N \times N$ identity matrix, $\text{diag}(v_i(t))$ is the diagonal matrix with elements $v_1(t), v_2(t), \dots, v_N(t)$ and B is the infection rate matrix with elements β_{ij} . The trivial, i.e. all-zero, solution of (3) indicates the absorbing state where all nodes are susceptible. The non-zero solution of V_∞ in (3), if exists, points to the existence of a metastable state with a non-zero fraction of infected nodes. Or else, the metastable state can be figured as 0 or not existing. We are interested in actually the metastable state in this paper.

2.2 The i.i.d. heterogeneous infection rates

In this paper, we keep the average infection rate to 1 and tune the recovery rate δ to control the effective infection rate τ . In the case of the i.i.d. heterogeneous infection rates, we aim to explore how the heterogeneous infection rates influence the accuracy of NIMFA when the variance of the infection rate varies. We choose the infection-rate distribution that is frequently observed in real-world and importantly the variance is tunable with a fixed mean so that we can systematically explore how the accuracy of NIMFA changes with the broadness of the i.i.d. infection rate.

We consider the log-normal distribution, of which we can keep the mean unchanged and tune the variance in a large range. The log-normal distribution [18]

$B \sim \text{Log-N}(\beta; \mu, \sigma)$, of which the probability density function (PDF) is, for $\beta > 0$

$$f_B(\beta; \mu, \sigma) = \frac{1}{\beta \sigma \sqrt{2\pi}} \exp\left(-\frac{(\ln \beta - \mu)^2}{2\sigma^2}\right) \quad (4)$$

has a power-law tail for a large range of β provided σ is sufficiently large. The log-normal distribution has been widely observed in real-world, where interaction frequencies between nodes are usually considered as infection rates. Wang et al. [21] find that by employing the log-normal distributed infection rates, their epidemic model can accurately fit the infection data of 2003 SARS; we also find that the infection rates in an airline network follow the log-normal distribution [15].

In [15], we find that, if the epidemic does not die out, the larger the variance of the i.i.d. infection rate is, the smaller the average fraction y_∞ of infected nodes is. We will show that this conclusion can actually explain the observation about how the accuracy of NIMFA changes with the variance of the i.i.d. infection rates at a given effective infection rate τ in this paper.

2.3 The correlated heterogeneous infection rates

For correlated heterogeneous infection rates, we build a correlated infection-rate scenario and a reference one. In the correlated infection-rate scenario, we assume

$$\beta_{ij} = c(d_i d_j)^\alpha \quad (5)$$

where d_i and d_j are the degree of node i and node j respectively, c is selected so that the average infection rate is 1 and α indicates the correlation strength. As discussed in [17], such a correlation between the infection rate and the nodal degree is motivated by the real-world datasets. In this case, the infection rate of each link is determined by the given network topology and α . For the reference scenario, we shuffle the infection rates from all the links as generated in the first scenario and redistribute them randomly to all the links. In this way, we keep the distribution of infection rates but effectively remove the correlation between the infection rates and nodal degrees. For simplicity, we name this reference scenario as the uncorrelated infection-rate scenario. Though the i.i.d. infection rates are also uncorrelated, we can tune the variance of the infection rate in the case of the i.i.d. infection rates while keeping the distribution and the mean of the infection rates. However, in the scenario of uncorrelated infection rates in this paper, the distribution of the infection rate changes with the parameter α , hence the variance of the heterogeneous infection rates cannot be systematically tuned.

A positive $\alpha > 0$ (or negative $\alpha < 0$), suggests a positive (or negative) correlation between infection rates and nodal degrees. Too large or small values of α could not be realistic. For example, [3, 9, 11] suggest that α is around 0.5 or 0.8 in their datasets. Hence, we select $\alpha = -0.25, -0.5, -1$ for the negative correlation and

$\alpha = 0.25, 0.5, 1$ for the positive correlation. Different values of α also offer the possibility to explore how NIMFA performs with different correlation strengths.

In this paper, we aim to understand how the correlation influences the accuracy of NIMFA by comparing the average fraction y_∞ of infected nodes obtained by NIMFA and the simulations of the exact SIS model. In [17], we explored the influence of the correlation between the infection rate and the nodal degree on the prevalence of epidemic, which can be used to partially explain the conclusions in this paper.

2.4 The network construction and simulations

As in our previous work [15, 17], we perform the continuous-time simulations of the SIS model. We consider both the scale-free (SF) and Erdős-Rényi (ER) models for different network topologies. The SF model has been used to capture the scale-free nature of degree distribution in real-world networks such as the Internet [5] and World Wide Web [1]: $\Pr[D = d] \sim d^{-\lambda}, d \in [d_{\min}, d_{\max}]$, where d_{\min} is the smallest degree, d_{\max} is the degree cutoff, and $\lambda > 0$ is the exponent characterizing the broadness of the distribution [2]. In real-world networks, the exponent λ is usually in the range [2, 3], thus we confine the exponent $\lambda = 2.5$ in this paper. We further employ the smallest degree $d_{\min} = 2$, the natural degree cutoff $d_{\max} = \lfloor N^{1/(\lambda-1)} \rfloor$ as in [6], and the size $N = 1000$. Hence, the average degree is approximately 4. The distribution of the degree of a random node in ER network is binomial: $\Pr[D = d] = \binom{N-1}{d} p^d (1-p)^{N-1-d}$ and the average degree is $E[D] = (N-1)p$. We consider the ER networks with $N = 1000$ and $E[D] = 4$.

Given a network topology and a recovery rate δ , we carry out 100 iterations. In each iteration, the networks are constructed as described above and the infection rates are generated as described in Section 2.2 and 2.3. Initially, 10% of the nodes are randomly infected. Then the infection and recovery processes of SIS model are simulated until the system reaches the metastable state where the fraction of infected nodes is nonzero and unchanged for a long time if the epidemic spreads out, or the fraction is zero if the epidemic dies out. The average fraction y_∞ of infected nodes is obtained over 100 iterations (no matter the epidemic dies out or not).

3 Effect of the heterogeneous infection rates

In this section, we first explore the accuracy of NIMFA with the i.i.d. infection rates, and particularly how NIMFA performs when the variance $\text{Var}[B]$ of the infection rate B varies. Then we explore the influence of the correlated infection rates on NIMFA.

3.1 The i.i.d. infection rates

We aim to understand the precision of NIMFA under different effective infection rates, different variances of infection rates and different network topologies: we set

the average infection rate to 1 and tune the recovery rate δ to control the effective infection rate τ ; we change the variance of infection rates which follow the log-normal distribution; we consider both ER and SF networks to represent different topologies. For each value of the variance of the infection rate, we obtain the average fraction y_∞ of infected nodes as a function of the effective infection rate τ for NIMFA by numerically solving (3) and compare with that by the continuous-time simulations. As shown in Fig. 1a, no matter what the variance of the infection rate is, the curve of y_∞ vs. τ obtained by NIMFA for ER networks is close to that obtained by simulations when the actual prevalence of the epidemic is high, i.e. the effective infection rate τ is large.

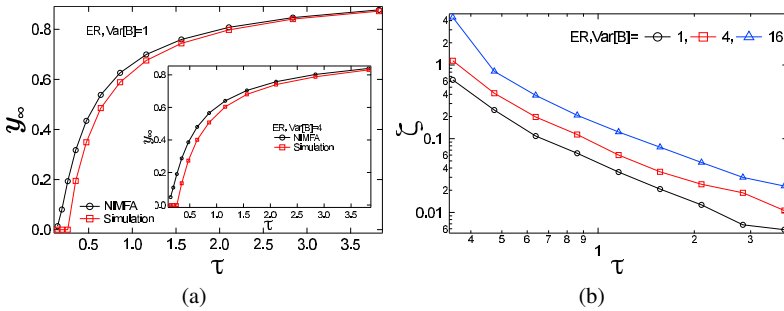


Fig. 1: (a) The average fraction y_∞ as a function of the effective infection rate τ and (b) the plot of the difference ζ as a function of the effective infection rate τ . The variances of the infection rates are 1 and 4 in the main figure and the inset respectively. All results are on ER networks.

In order to quantify the difference between the two curves obtained by NIMFA and simulations, we define the variable ζ :

$$\zeta(\tau) = \frac{|y_{\infty,N}(\tau) - y_{\infty,S}(\tau)|}{y_{\infty,S}(\tau)} \tag{6}$$

where $y_{\infty,N}(\tau) > 0$ and $y_{\infty,S}(\tau) > 0$ denote the average fraction of infected nodes obtained by NIMFA and simulations respectively. The larger the value of $\zeta(\tau)$ is, the less accurate NIMFA is at the corresponding τ .

In Fig. 1b, the plot of ζ vs. τ is shown for ER networks. We find that, for a given effective infection rate τ , NIMFA becomes less accurate when the variance of the i.i.d. heterogeneous infection rates increases. This observation can be to a large extent explained by: 1) our finding in Fig. 1a that NIMFA is more accurate when the prevalence is higher; 2) that given an effective infection rate τ a smaller variance of the i.i.d. infection rates leads to a higher prevalence [15]. We observe the same in SF networks, and the figures, which can be found in [16], are not shown here due to the page limit.

We further explore how the variance of the infection rates influences the accuracy of NIMFA if the actual prevalence $y_{\infty,S}(\tau)$ of epidemic is similar. We plot the variable ζ in (6) as a function of the actual average fraction of infected nodes obtained by simulations in Fig. 2. We find that though it is less evident for ER networks in Fig. 2a, the difference ζ in (6) is actually larger if the variance of the infection rate is larger as shown in Fig. 2b for SF networks when the prevalence is the same. Hence, the higher heterogeneity, i.e. the larger variance, of the i.i.d. infection rates tends to lower down more the accuracy of NIMFA. Overall, we conclude that the prevalence of the epidemic mainly affects the accuracy of NIMFA, i.e. the higher the prevalence is, the more accurate NIMFA tends to be, and given the same prevalence, a larger variance of the i.i.d. infection rates tends to lower down the accuracy of NIMFA.

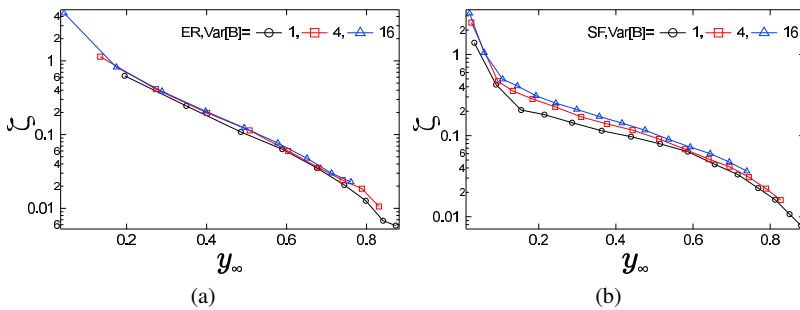


Fig. 2: The plot of the difference ζ as a function of the average fraction y_{∞} obtained by simulations for (a) ER networks and (b) SF networks.

3.2 The correlated infection rate

In this subsection, we aim to understand how the correlation between the infection rate and the nodal degree as shown in (5) influences the accuracy of NIMFA. We first employ ER networks as an example and discuss the case when the correlation is positive. Afterwards we explore the influence of the negative correlation.

As mentioned in Section 2.3, we build the scenario of uncorrelated infection rates as a reference to study the influence of the correlation between the infection rate and the nodal degree by shuffling the infection rates from all the links as generated in the scenario of correlated infection rates and redistributing them randomly to all the links. As shown in Fig. 3a, we compare the difference ζ between NIMFA and simulations in the scenario of uncorrelated and correlated infection rates for both $\alpha = 0.25$ and $\alpha = 1$, and find that ζ is smaller in the scenario of correlated infection rates, i.e. NIMFA is more accurate at a given effective infection rate τ when the correlation between the infection rate and the nodal degree is positive comparing to the scenario of uncorrelated infection rates. The observations are also consistent

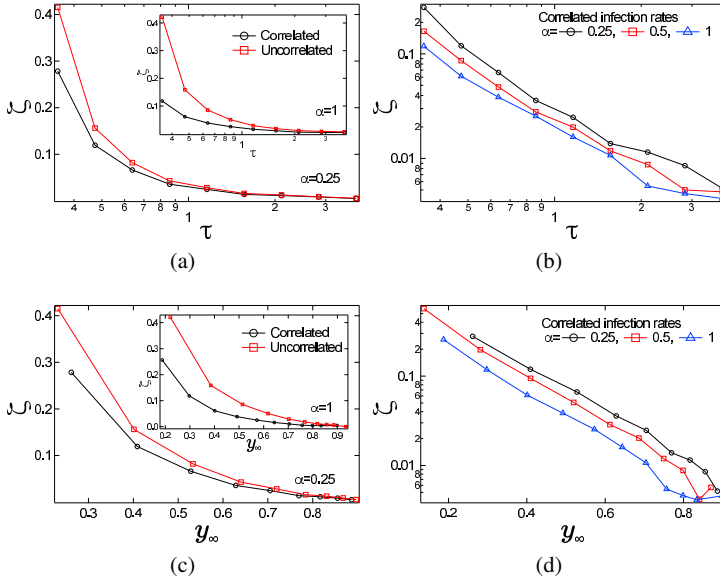


Fig. 3: The plot of the difference ζ as a function of (a) the effective infection rate τ or (c) the average fraction y_∞ of infected nodes obtained by simulations in the scenarios of uncorrelated and correlated infection rates for $\alpha = 0.25$ (the main figure) and $\alpha = 1$ (the inset). The plot of the difference ζ as a function of (b) the effective infection rate τ or (d) the actual average fraction y_∞ of infected nodes in the scenario of correlated infection rates where different values of α are considered.

with our conclusion that NIMFA is more accurate when the prevalence is higher: the positive correlation tends to increase the average fraction of infected nodes [17], and thus the accuracy of NIMFA, when the effective infection rate τ is small; however, when the effective infection rate τ is large, though the positive correlate may lower down a bit the average fraction y_∞ of infected nodes, the prevalence in both scenarios is high, i.e. NIMFA is relatively accurate, and the difference of the accuracy of NIMFA in the two scenarios is not obvious. As the correlation strength α increases in Fig. 3b, the difference ζ decreases at a given τ . That is to say, NIMFA tends to be more accurate when the positive correlation becomes stronger.

We further consider the influence of the positive correlation on the accuracy of NIMFA when the prevalence is the same. The plots of the difference ζ as a function of the average fraction y_∞ of infected nodes are shown in Fig. 3c and Fig. 3d. Given the prevalence of epidemic, the positive correlation is more likely to increase the precision of NIMFA and the stronger the correlation is the more accurate NIMFA is. We observe the same on SF networks which is though not shown here.

Regarding to the influence of the negative correlation between the infection rate and the nodal degree on the accuracy of NIMFA, we compare the variable ζ in the scenario of correlated and uncorrelated infection-rate scenario with $\alpha = -1$

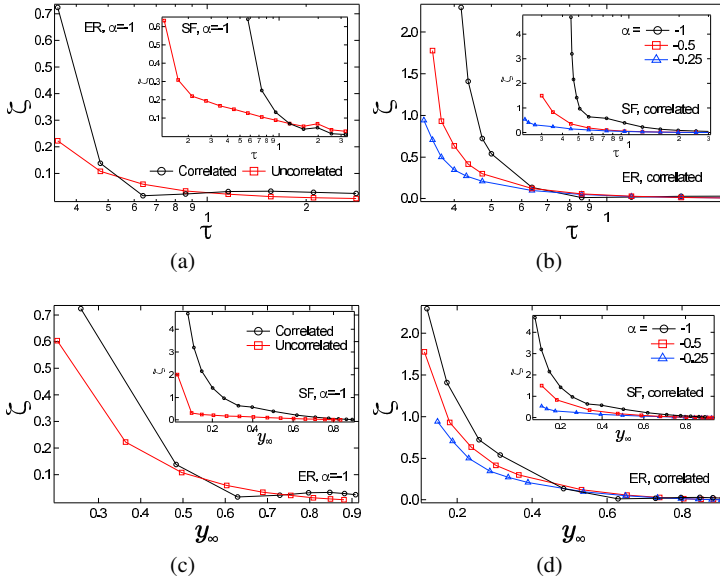


Fig. 4: The plot of the difference ζ as a function of (a) the effective infection rate τ or (c) the actual average fraction y_∞ of infected nodes in the scenarios of uncorrelated and correlated infection rates for $\alpha = -1$. The plot of the difference ζ as a function of (b) the effective infection rate τ or (d) the actual average fraction y_∞ of infected nodes in the scenario of correlated infection rates where different values of α are considered.

for both ER and SF networks as shown in Fig. 4a. We find that, in general, the negative correlation significantly decreases the accuracy of NIMFA when the effective infection rate τ is small but may slightly increase that when τ is large. Moreover, NIMFA becomes less accurate when the negative correlation is stronger as shown in Fig. 4b. As mentioned in Section 2.3, the negative correlation tends to decrease the prevalence when the effective infection rate τ is small while increase the prevalence when τ is large. Hence, the influence of prevalence on the precision of NIMFA could largely explain our observations here.

When the prevalence of epidemic is the same, the influence of the negative correlation on NIMFA’s accuracy is shown in Fig. 4c and Fig. 4d. We find that, in general, 1) NIMFA is less accurate with the negative correlation comparing to the uncorrelated scenario especially when the prevalence is low as shown in Fig. 4c; 2) NIMFA becomes even less accurate if the negative correlation becomes stronger as shown in Fig. 4d.

4 Real-world network

In this section, we choose the airline network from the real world as an example to illustrate how its heterogeneous infection rates affect the accuracy of NIMFA of SIS epidemics on the network.

In the airline network, the nodes are the airports, the link between two nodes indicates that there's at least one flight between these two airports, and the infection rate along a link is the number of flights between the two airports. We construct this network and its infection rates from the dataset of openFlights¹. As shown in [17], the airline network possess roughly a power-law degree distribution. The heterogeneous infection rates from the dataset are normalized by the average so that the average is 1. We compare the difference ζ between NIMFA and the simulations of the exact SIS model in three scenarios: 1) the network is equipped with its normalized original heterogeneous infection rates (correlated) as given in the dataset; 2) the network is equipped with the infection rates in the normalized original dataset but randomly shuffled (uncorrelated); 3) the network is equipped with a constant infection rate (homogeneous) which equals to 1. The original heterogeneous infection rate between a pair of nodes are approximately correlated with the degrees of the two nodes as the relationship (5), and the parameter $\alpha \approx 0.14$ indicates a positive correlation [17].

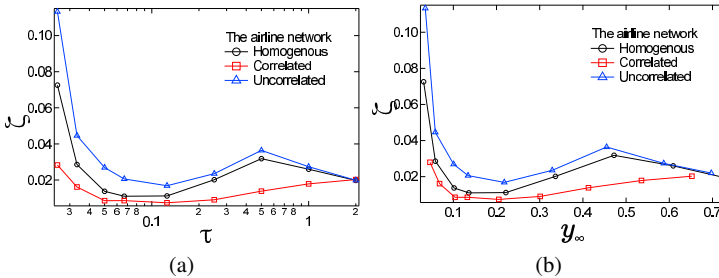


Fig. 5: The plot of the difference ζ as a function of (a) the effective infection rate τ and (b) the average fraction y_∞ of infected nodes obtained by simulations in the airline network with different scenarios of infection rates.

We show the difference ζ as a function of the effective infection rate τ in Fig. 5a for the 3 scenarios defined as above. We find that NIMFA is generally more accurate when the effective infection rate τ is larger, i.e. the prevalence of epidemic is high. The variable ζ is smaller in the scenario of homogeneous infection rates than uncorrelated infection rates with any effective infection rate. This is because the i.i.d. infection rates with a non-zero variance tends to decrease the prevalence, and thus lower down the accuracy of NIMFA at a given effective infection rate τ . NIMFA is more accurate with the positive correlation by comparing the difference ζ in the scenario of correlated infection rates and uncorrelated infection rates. Furthermore, Fig. 5b

¹ <http://openflights.org/data.html>

shows that, given the same actual prevalence, i.e. the average fraction y_∞ of infected nodes obtained by simulations, NIMFA is more accurate: 1) in the homogeneous scenario than in the uncorrelated scenario; 2) in the correlated scenario than in the uncorrelated scenario. All the observations agree with our previous observations and explanations about how the heterogeneous infection rate influences the accuracy of NIMFA in network models.

5 Conclusion

In this paper, we study how the heterogeneous infection rates affect the accuracy of NIMFA – an advanced mean-field approximation of SIS model that takes the underlying network topology into account. By comparing NIMFA with the continuous-time simulations of the exact SIS model at a given effective infection rate τ , we find that the prevalence of epidemic could largely characterize the accuracy of NIMFA which is reflected in two aspects: 1) NIMFA is generally more accurate when the effective infection rate τ is larger, i.e. the prevalence of epidemic is higher; 2) when the variance of the i.i.d. infection rates or the correlation between the infection rate and the nodal degree decreases the prevalence at a given τ , NIMFA tends to become less accurate as well. Moreover, we also explore the influence of the heterogeneous infection rates on the accuracy of NIMFA at a given prevalence, i.e. when the average fraction y_∞ of infected nodes obtained by simulations is given. Regarding to the i.i.d. heterogeneous infection rates, the accuracy of NIMFA tends to decrease as the variance of infection rates increases. In the scenario of correlated infection rates, the positive correlation between the nodal degree and the infection rate is more likely to increase the accuracy of NIMFA whereas the negative correlation tends to lower down the accuracy especially when the effective infection rate τ is small. Note that we discuss the conditions when NIMFA is accurate but the cases where NIMFA is far from the simulations are still unexplored. Our work sheds light on the conditions when the mean-field approximation of the SIS model with heterogeneous infection rates is accurate.

References

- [1] Albert, R., Jeong, H., Barabási, A.L.: Internet: Diameter of the world-wide web. *Nature* **401**(6749), 130–131 (1999)
- [2] Barabási, A.L., Albert, R.: Emergence of scaling in random networks. *Science* **286**(5439), 509–512 (1999)
- [3] Barrat, A., Barthélemy, M., Pastor-Satorras, R., Vespignani, A.: The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences of the United States of America* **101**(11), 3747–3752 (2004)
- [4] Buono, C., Vazquez, F., Macri, P., Braunstein, L.: Slow epidemic extinction in populations with heterogeneous infection rates. *Physical Review E* **88**(2), 022,813 (2013)
- [5] Caldarelli, G., Marchetti, R., Pietronero, L.: The fractal properties of internet. *EPL (Europhysics Letters)* **52**(4), 386 (2000)

- [6] Cohen, R., Erez, K., ben Avraham, D., Havlin, S.: Resilience of the internet to random breakdowns. *Physical Review Letters* **85**, 4626–4628 (2000). DOI 10.1103/PhysRevLett.85.4626. URL <http://link.aps.org/doi/10.1103/PhysRevLett.85.4626>
- [7] Li, C., van de Bovenkamp, R., Van Mieghem, P.: Susceptible-infected-susceptible model: A comparison of n-intertwined and heterogeneous mean-field approximations. *Phys. Rev. E* **86**(2), 026,116 (2012)
- [8] Li, D., Qin, P., Wang, H., Liu, C., Jiang, Y.: Epidemics on interconnected lattices. *EPL (Europhysics Letters)* **105**(6), 68,004 (2014). URL <http://stacks.iop.org/0295-5075/105/i=6/a=68004>
- [9] Li, W., Cai, X.: Statistical analysis of airport network of china. *Physical Review E* **69**(4), 046,106 (2004)
- [10] Liu, M., Li, D., Qin, P., Liu, C., Wang, H., Wang, F.: Epidemics in interconnected small-world networks. *PLoS one* **10**(3), e0120,701 (2015)
- [11] Macdonald, P., Almaas, E., Barabási, A.L.: Minimum spanning trees of weighted scale-free networks. *EPL (Europhysics Letters)* **72**(2), 308 (2005)
- [12] Pastor-Satorras, R., Castellano, C., Van Mieghem, P., Vespignani, A.: Epidemic processes in complex networks. *arXiv preprint arXiv:1408.2701* (2014)
- [13] Pastor-Satorras, R., Vespignani, A.: Epidemic dynamics and endemic states in complex networks. *Physical Review E* **63**(6), 066,117 (2001)
- [14] Pastor-Satorras, R., Vespignani, A.: Epidemic spreading in scale-free networks. *Physical Review Letters* **86**(14), 3200 (2001)
- [15] Qu, B., Wang, H.: SIS epidemic spreading with heterogeneous infection rates. *arXiv preprint arXiv:1506.07293* (2015)
- [16] Qu, B., Wang, H.: The accuracy of mean-field approximation for susceptible-infected-susceptible epidemic spreading. *arXiv preprint arXiv:1609.01105* (2016)
- [17] Qu, B., Wang, H.: SIS epidemic spreading with correlated heterogeneous infection rates. *arXiv preprint arXiv:1608.07327* (2016)
- [18] Van Mieghem, P.: Performance analysis of communications networks and systems. Cambridge University Press (2014)
- [19] Van Mieghem, P., Omic, J., Kooij, R.: Virus spread in networks. *IEEE/ACM Transactions on Networking* **17**(1), 1–14 (2009)
- [20] Wang, H., Li, Q., D’Agostino, G., Havlin, S., Stanley, H.E., Van Mieghem, P.: Effect of the interconnected network structure on the epidemic threshold. *Physical Review E* **88**(2), 022,801 (2013)
- [21] Wang, W., Wu, Z., Wang, C., Hu, R.: Modelling the spreading rate of controlled communicable epidemics through an entropy-based thermodynamic model. *Sci. Sin.-Phys. Mech. Astron.* **56**(11), 2143 (2013). DOI 10.1007/s11433-013-5321-0
- [22] Yang, Z., Zhou, T.: Epidemic spreading in weighted networks: an edge-based mean-field solution. *Physical Review E* **85**(5), 056,106 (2012)

Die-out Probability in SIS Epidemic Processes on Networks

Qiang Liu and Piet Van Mieghem

Abstract An accurate approximate formula of the die-out probability in a SIS epidemic process on a network is proposed. The formula contains only three essential parameters: the largest eigenvalue of the adjacency matrix of the network, the effective infection rate of the virus, and the initial number of infected nodes in the network. The die-out probability formula is compared with the exact die-out probability in complete graphs, Erdős-Rényi graphs, and a power-law graph. Furthermore, as an example, the formula is applied to the N -Intertwined Mean-Field Approximation, to explicitly incorporate the die-out.

1 Introduction

The SIS epidemic process models spreading phenomena of information or viruses on networks [7]. In a network, each node has two states: susceptible and infected. A Bernoulli random variable $X_j(t) \in \{0, 1\}$ denotes the state of each node, where $X_j(t) = 0$ means that node j is susceptible and $X_j(t) = 1$ indicates that node j is infected at time t . An infected node can infect its susceptible neighbors with a infection rate β by changing the susceptible neighbor nodes into infected nodes, and each infected node is cured and becomes a susceptible node with a curing rate δ . If the infection and curing processes are Poisson processes, the SIS epidemic model is Markovian, where the sojourn times in the infected and susceptible state are exponentially distributed. The governing equation of a node j in the Markovian SIS epidemic process in an unweighted and undirected network with N nodes, represented by an $N \times N$ symmetric adjacency matrix A , is [10, p. 449]

Qiang Liu (e-mail: Q.L.Liu@TuDelft.nl)✉ · Piet Van Mieghem (e-mail: P.F.A.VanMieghem@TuDelft.nl)

Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, P.O Box 5031, 2600 GA Delft, The Netherlands

$$\frac{dE[X_j(t)]}{dt} = -\delta E[X_j(t)] + \beta \sum_{k=1}^N a_{kj} E[X_k(t)] - \beta \sum_{k=1}^N a_{kj} E[X_j(t)X_k(t)] \quad (1)$$

The epidemic threshold τ_c of the SIS epidemic process implies that, if the effective infection rate $\tau = \beta/\delta > \tau_c$, the virus will spread over the network for a very long time, and if $\tau < \tau_c$, the number of infected nodes decreases exponentially fast after sufficiently long time [7, 11]. There is an approximate value [14] and lower bound [13] of the epidemic threshold $\tau_c > \tau_c^{(1)} = 1/\lambda_1$, where λ_1 is the largest eigenvalue of the adjacency matrix A . In this paper, the threshold $\tau_c^{(1)}$ is referred to as the N -Intertwined Mean Field Approximation (NIMFA) threshold, where the superscript (1) in $\tau_c^{(1)}$ refers to the fact that NIMFA is a first order mean-field approximation [13].

The structure of this paper is organized as follows. Section 2 introduces the relation between the prevalence (2) and the average fraction of infected nodes conditioned to the survival of the virus. Clearly, the virus die-out probability plays a key role. Section 3 proposes an accurate approximate formula (6) for the die-out probability in the metastable state of the SIS epidemic process. Figure 1 and Fig. 2 demonstrate the accuracy and the limitation of (6) in complete graphs, Erdős-Rényi graphs, and power-law graphs. Finally, we apply formula (6) to correct the NIMFA prevalence (8) as shown in Fig. 3.

2 The Prevalence and the Die-out Probability

The prevalence $y(t)$ of a SIS epidemic process is the expected fraction of infected nodes at time t ,

$$y(t) = E[S(t)] \quad (2)$$

where $S(t) = \frac{1}{N} \sum_{j=1}^N X_j(t)$ is the fraction of infected nodes. The prevalence in the exact Markovian epidemic process after infinitely long time tends to zero, where the absorbing state is reached. Before the virus dies out, virus may exist in networks for a very long time [2, 9]. In the metastable state, the prevalence $y(t)$ changes very slowly and there is a balance between the infection and curing processes. We confine ourselves to the time region $[0, t_{max}]$ that the prevalence $y(t_{max}) \neq 0$, and the prevalence is approximately equal at every time $t \in [t_m, t_{max}]$, where t_m is the time that the SIS process reaches metastable state. However, for one realization of the epidemic process, we cannot expect that the fraction of infected nodes oscillates around the level of the prevalence $y(t)$ with time t , because the prevalence $y(t)$ is the average over all possible realizations including the die-out realizations. In real observed diseases, the virus has not died out yet, so that the fraction of infected population is positive. So, there are actually two kinds of average: the average over all possible realizations (prevalence), and the average over the realizations conditioned to the survival of the virus. To prevent confusion, the fraction of infected nodes under the condition that the virus survives at time t is denoted by a random variable $\tilde{S}(t)$ in this paper. Consequently, we have $\Pr[\tilde{S}(t) = i/N] = \Pr[S(t) = i/N | S(t) \neq 0]$

for $\tilde{S}(t) \in \{1/N, 2/N, \dots, 1\}$ while $S(t) \in \{0, 1/N, 2/N, \dots, 1\}$. The removal of the absorbing state [3] or the assumption that the virus survives is associated with the quasi-stationarity or metastability of the SIS process [8]. The expectation of $\tilde{S}(t)$ of an epidemic process in a network with N nodes is

$$E[\tilde{S}(t)] = \sum_{i=1}^N \frac{i}{N} \Pr[\tilde{S}(t)] = \sum_{i=1}^N \frac{i}{N} \Pr \left[S(t) = \frac{i}{N} \mid S(t) \neq 0 \right]$$

With the definition of the conditional probability,

$$\begin{aligned} \Pr \left[S(t) = \frac{i}{N} \mid S(t) \neq 0 \right] &= \frac{\Pr \left[\left\{ S(t) = \frac{i}{N} \right\} \cap \{ S(t) \neq 0 \} \right]}{\Pr[S(t) \neq 0]} \\ &= \frac{\Pr \left[S(t) = \frac{i}{N} \right]}{\Pr[S(t) \neq 0]} \quad \text{provided } i > 0 \end{aligned}$$

we have

$$E[\tilde{S}(t)] = \frac{1}{\Pr[S(t) \neq 0]} \sum_{i=0}^N \frac{i}{N} \Pr \left[S(t) = \frac{i}{N} \right] = \frac{E[S(t)]}{\Pr[S(t) \neq 0]}$$

Since $\Pr[S(t) \neq 0] = 1 - \Pr[S(t) = 0]$, the prevalence can be written as

$$y(t) = \tilde{y}(t) (1 - \Pr[S(t) = 0]) \tag{3}$$

where $\tilde{y}(t) = E[\tilde{S}(t)]$. Equation (3) shows the relation between the prevalence $y(t)$ and the average fraction $\tilde{y}(t)$ of infected nodes under the condition that the virus survives, where the die-out probability $\Pr[S(t) = 0]$ is essential. Both the prevalence $y(t)$ and the virus die-out probability $\Pr[S(t) = 0]$ are difficult to compute analytically in general graphs.

The Markovian SIS epidemic process on the complete graph K_N is a birth-and-death process [3, 10]. The states $\{0, 1, \dots, N\}$ of the birth-and-death process are the number of infected nodes, where 0 is the absorbing state or overall-healthy state. Therefore, the die-out probability $\Pr[S(t) = 0]$ can be obtained by solving the birth-and-death process,

$$(\mathbf{s}'(t))^T = \mathbf{s}^T(t) \mathbf{Q} \tag{4}$$

where \mathbf{Q} is the infinitesimal generator of the birth-and-death Markov chain, and $\mathbf{s}^T(t) = [s_0(t), \dots, s_N(t)]$ is the state probability vector with each element $s_i(t) = \Pr[S(t) = i/N]$ for $0 \leq i \leq N$, and $s_0(t) = \Pr[S(t) = 0]$.

The die-out probability $\Pr[S(t) = 0]$ of SIS epidemic process in complete graphs also equals the gambler's ruin probability [10, p. 231] as shown in the Appendix,

$$\mu_n = \frac{\sum_{j=0}^{N-n-1} j! \tau^j}{\sum_{j=0}^{N-1} j! \tau^j} \quad (5)$$

Different from solving (4), Eq. (5) only applies to the metastable state and cannot be used to calculate the die-out probability at an arbitrary time t . As demonstrated in the Appendix, Eq.(5) upper bounds the actual die-out probability, because (5) assumes that the virus wins only when it infects all N nodes in a finite time before dying out.

3 The Die-out Probability: an Accurate Approximation

Apart from solving (4) or employing the gambler's ruin formula (5), in this section we propose a novel approximate formula of the virus die-out probability in the metastable state.

We assume that the prevalence $y(t)$ is approximately constant in the metastable state. Relation (3) then indicates that the die-out probability is also approximately constant. In the metastable state, we then find that the virus die-out probability in a sufficiently large graph is approximately

$$\Pr[S(t_m) = 0] \approx \frac{1}{x^n}, \quad \text{with } x \geq 1 \quad (6)$$

where $S(t_m)$ denotes the fraction of infected nodes of the SIS epidemic process in the metastable reached at time t_m , $x = \tau/\tau_c^{(1)} = \lambda_1 \tau$ is the normalized effective infection rate of the virus, and n is the number of initially infected nodes. The situation $x < 1$ is not considered, because the infection rate is below the threshold and the SIS process dies out before reaching the metastable state. In addition, $1/x > 1$ cannot represent a probability. As the first order NIMFA threshold $\tau_c^{(1)} = 1/\lambda_1$ is a lower bound of the actual threshold τ_c , the prevalence $y(t)$ decreases exponentially fast for sufficiently large time [11] when $x \leq 1$, and the virus die-out probability tends to 1. Also, the accuracy of formula (6) is related to the accuracy of the NIMFA threshold $\tau_c^{(1)} = 1/\lambda_1$. For example, if the effective infection rate is below the real threshold and $\tau_c^{(1)} < \tau < \tau_c$, formula $1/x^n < 1$, but the virus dies out within finite time with the probability tending to 1. In the Appendix, an analytically approach to (6) from the gambler's ruin probability (5) in complete graphs is presented.

By introducing the normalized effective infection rate $x = \tau/\tau_c^{(1)} = \tau\lambda_1$ into (6), the network topology—the largest eigenvalue of the adjacency matrix λ_1 —is reflected. Formula (6) is simple, and only three essential parameters are involved: the spectral radius λ_1 , the virus spreading ability τ , and the initially infected number of nodes n . If a few nodes are infected and the infection rate is above the threshold $x > 1$, then formula (6), which is equivalent to $\Pr[S(t_m) = 0] \approx e^{-n \log x}$, shows that the network will experience a disease outbreak, because the die-out probability decreases exponentially fast with n above the epidemic threshold ($\log x > 0$).

In the sequel, we compare (6) and the die-out probability $\Pr[S(t) = 0]$ obtained via simulations. The curing rate of all the calculations and simulations below is $\delta = 1$.

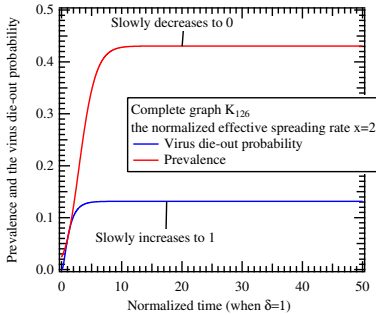
3.1 Complete Graphs

After solving the epidemic process (4) in the complete graph K_{126} with effective infection rate $\tau = 0.016$, Fig. 1a shows the prevalence $y(t)$ and the die-out probability $\Pr[S(t) = 0]$ as an example. The metastable state is reached approximately at time $t = 10$ and hereafter, and the prevalence $y(t)$ keeps steady. Also, the die-out probability $\Pr[S(t) = 0]$ becomes approximate constant earlier from $t = 5$. The prevalence $y(t)$ decreases slowly to 0 after an infinitely long time [2, 9], and correspondingly, the die-out probability increases to 1. At $t = 45$ in the metastable state, the number of die-out realizations of the SIS epidemic simulation and the solution of the Markov chain Eq. (4) are recorded and shown in Fig. 1b, 1c, and 1d. The simulation results in Fig. 1b and 1c are obtained by the SSIS simulator [1] which applies a Gillespie-like algorithm [4], and 10^6 realizations of the Markovian epidemic process are simulated. By counting the number of realizations which have zero infected nodes at $t = 45$, the die-out probability is obtained.

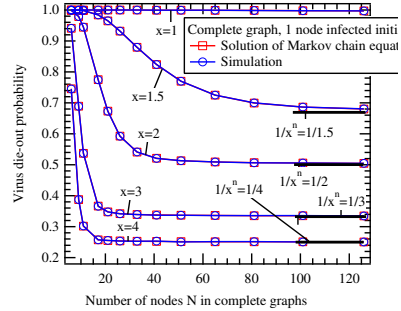
Figure 1b and 1c illustrate that, our simulation results match with the computation of the birth-and-death process (4). To avoid redundancy, we omit the simulation results in Fig. 1d. From Fig. 1b, the die-out probability at $t = 45$ is approximately 1 corresponding to formula (6), when the normalized effective infection rate $x = 1$. Also, if $x = 1$, the infection rate is below the threshold, and no matter how many nodes are infected initially, the prevalence $y(t)$ decreases exponentially fast for sufficiently large time. The mean-field approximations are usually not accurate around threshold [6], which is also verified by Eq. (3) when $x = 1$ and the die-out probability $\Pr[S(t_m) = 0] = 1$. For a different number of initially infected nodes n , Fig. 1 shows that the virus die-out probabilities converge to the concise formula (6) fast with the network size N . Furthermore, the larger the normalized effective infection rate x is, the faster the probabilities convergence towards (6).

3.2 General Graphs

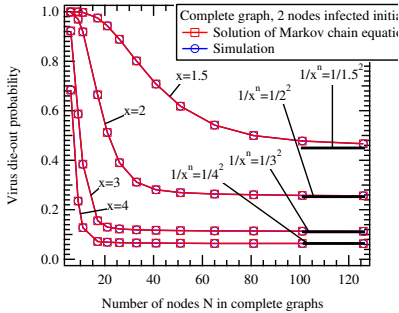
For general graphs, it is infeasible to obtain the virus die-out probability by solving the differential equations of Markov chain, because the number of equations is 2^N . However, it is still possible to obtain the virus die-out probability efficiently by simulation. We construct three Erdős-Rényi (ER) graphs $G_p(N)$ with the network size $N = 100$ and the link generation probability $p = 0.9, 0.5$, and 0.1 , respectively. The epidemic process is simulated on the ER graphs by randomly choosing the initially infected nodes. For every normalized infection rate x and every number of initially infected nodes n , 10^4 realizations are simulated. Fig. 2a, 2b, and 2c give the the comparison between the die-out probabilities and formula (6) for the number of initially infected nodes $n = 1, 2, 3$. Formula (6) is accurate in the general ER graphs,



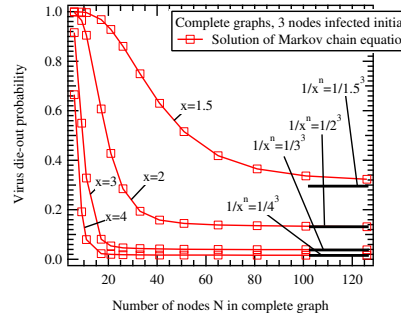
(a) The virus die-out probability and the prevalence of epidemic process in complete graph K_{126} . Initially 3 nodes are infected. This figure shows a clearly metastable state region.



(b) The die-out probabilities from simulation of the SIS epidemic process and calculation of the birth-and-death process are shown with 1 initially infected node. With the increase of network size N , the die-out probabilities converge to the simple formula: $1/x^n$.



(c) With 2 nodes infected initially, this figure verifies (6) as Fig. 1b with simulation and calculation results.

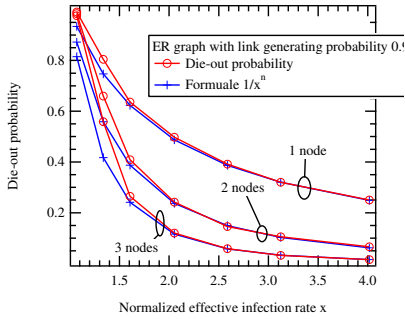


(d) With 3 nodes infected initially, this figure shows the calculation results of (4) as Fig. 1b and 1c.

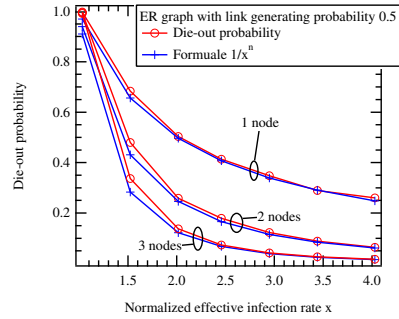
Fig. 1: The virus die-out probability in complete graphs.

especially when the normalized effective infection rate x is large. The accuracy of formula (6) decreases with decreasing link generation probability p in ER graphs $G_p(N)$.

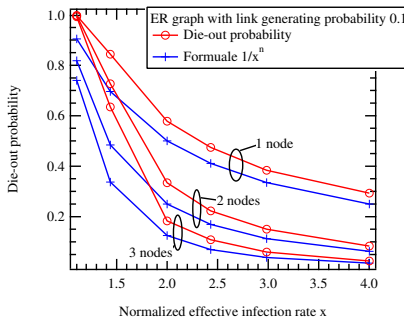
The die-out probability of the SIS epidemic process in a power-law graph is presented in Fig. 2d with 10^5 realizations, and formula (6) shows its limitation. The power law graph has $N = 1000$ nodes, and the degree distribution is $\text{Pr}[k] \sim k^{-2.6}$. Fig. 2d exhibits that the die-out probability is almost 1 when the normalized effective rate is around 2, which also indicates that the real epidemic threshold in the power-law graph is much larger than the NIMFA threshold $1/\lambda_1$. The inaccuracy of formula (6) is affected by the inaccuracy of the NIMFA threshold as mentioned above.



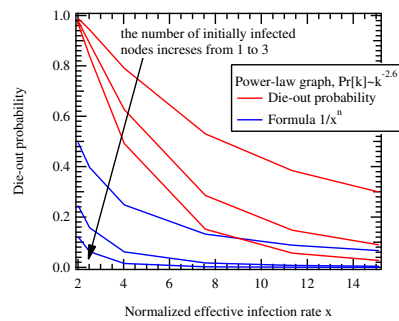
(a) The virus die-out probability of the SIS epidemic process in an ER graph with the link generation probability 0.9. The virus spreads starts from 1, 2, or 3 nodes initially.



(b) The die-out probability of the SIS epidemic process in another ER graph with the link generation probability 0.5.



(c) The die-out probability of the SIS epidemic process in another ER graph with the link generation probability only 0.1.



(d) The die-out probability of the SIS epidemic process in a power-law graph.

Fig. 2: The virus die-out probability in ER graphs and a power-law graph with different number of initially infected nodes.

The simulations seem to indicate that formula (6) is always smaller than the actual die-out probability, which may be attributed to the fact that the NIMFA threshold always lower bounds the actual threshold in any network.

3.3 NIMFA: Corrected for Die-out

The mean-field approximation methods are usually not accurate when the initial number of infected nodes is small, because the prevalence obtained by mean-field approximations will generally converge to fixed value due to the existence a steady state, no matter what the initial condition is. When a small number of nodes is initially infected, the die-out probability is relatively large. In this section, we will discuss the accuracy of NIMFA as an example. Previously, the accuracy of NIMFA has been

studied from a network topology viewpoint [12], but in this section, we focus on the influence of the initial condition.

NIMFA [13] reduces the computation complexity of a Markovian epidemic process by assuming independency between the state $X_j(t)$ of node j and the state $X_k(t)$ of node k , which closes the governing Eq. (1)

$$\frac{dv_j(t)}{dt} = -\delta v_j(t) + \beta \sum_{k=1}^N a_{kj} v_k(t) - \beta \sum_{k=1}^N a_{kj} v_j(t) v_k(t) \quad (7)$$

where $v_j(t)$ denotes the NIMFA infection probability of node j at time t . The NIMFA prevalence is similarly derived as

$$y^{(1)}(t) = \frac{1}{N} \sum_{j=1}^N v_j(t) \quad (8)$$

The NIMFA prevalence $y^{(1)}(t)$ decreases exponentially fast to 0 when the infection rate is below the NIMFA threshold $\tau \leq \tau_c^{(1)}$. If the initial condition $y^{(1)}(0) \neq 0$, the NIMFA prevalence $y^{(1)}(t)$ converges to a non-zero value when $\tau > \tau_c^{(1)}$, which is proved in [5]. Thus, NIMFA is conditioned to the case where the virus in the epidemic process will not die-out, and the absorbing state is removed when $y^{(1)}(0) \neq 0$. Based on (6), we propose an approximate virus surviving probability function at the time t as

$$f(t) = 1 - \frac{1}{x^n} + \frac{1}{x^n} e^{-\lambda_1 t} \quad (9)$$

Equation (9) is motivated as follows. At time $t = 0$ and $y^{(1)}(t) \neq 0$, the virus surviving probability is 1 and $f(0) = 1$, because a curing event happens with zero probability, when the time interval is 0. Next, simulations seem to indicate that the virus die-out probability decreases exponentially fast to $1/x^n$ in metastable state with a rate λ_1 .

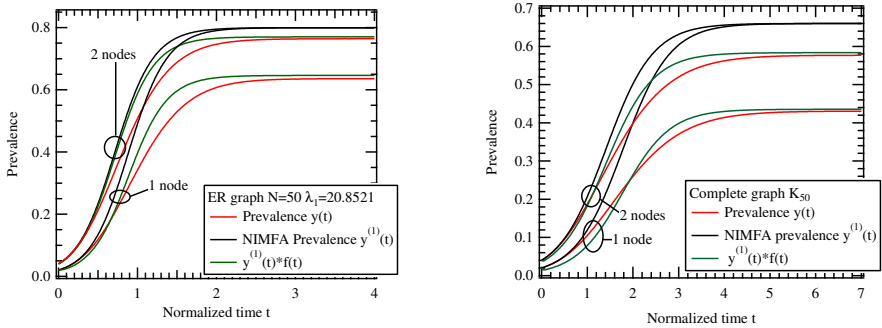
To incorporate the die-out, the NIMFA prevalence can be corrected by applying (3)

$$y(t) \approx y^{(1)}(t) f(t) \quad (10)$$

Figure 3 presents the prevalence and the approximation (10) of the SIS epidemic process in the complete graph K_{50} and the random generated ER graph in Sec 3.2. Starting from one or two infected nodes, NIMFA fails to predict the prevalence. The steady state of NIMFA is independent of the initial conditions. Fortunately, (10) seems a good approximation at the initial stage of the SIS epidemic process.

4 Conclusion

In this paper, we discuss the virus die-out probability, which is the probability that the SIS Markovian epidemic process reaches the absorbing state. The importance of the virus die-out probability lies in that it connects the virus spreading phenomena omitting die-out and the exact Markovian model with an absorbing state. Furthermore,



(a) SIS epidemic process in ER graph with network size $N = 50$. The effective infection rate is $\tau = 0.25$

(b) SIS epidemic in complete graph K_{50} with effective infection rate $\tau = 0.06$

Fig. 3: Comparison of NIMFA and the prevalence. The normalized time is the time scale when the curing rate $\delta = 1$ and the prevalence is obtained by averaging 10^6 realizations.

we propose an approximate formula (6) of the virus die-out probability, which only contains three essential parameters: the largest eigenvalue of adjacency matrix λ_1 (the topology parameter), the effective infection rate τ (the spreading ability parameter), and the number of initially infected node n (the initial condition parameter). If a few nodes are infected, then formula (6) indicates that the virus will almost surely cause a disease outbreak when the infection rate is above the threshold, irrespective of the network size N . However, the accuracy of formula (6) also depends on the accuracy of the NIMFA epidemic threshold $1/\lambda_1$. Based on formula (6), an approximate virus surviving probability function (9) is proposed. We also discuss the correction for NIMFA.

Acknowledgements Q. Liu would like to thank the support from China Scholarship Council.

Appendix

In the gambler’s ruin problem, the goal of the virus is to infect a certain number of nodes and to successfully reach the metastable state. If the virus cannot achieve the goal, the virus loses the game and dies out in the network. The analytic solution of the gambler’s ruin probability of a birth-and-process, which gives the probability μ_n that the virus dies out before infecting all N nodes in a finite time starting from an arbitrary number of infected nodes n , equals [10, p. 231],

$$\mu_n = \frac{\sum_{k=n}^{N-1} \prod_{m=1}^k \frac{1}{(N-m)\tau}}{1 + \sum_{k=1}^{N-1} \prod_{m=1}^k \frac{1}{(N-m)\tau}} \tag{11}$$

First, we evaluate the expression (11). Since

$$\prod_{m=1}^k \frac{1}{(N-m)\tau} = \frac{1}{\tau^k} \frac{(N-k-1)!}{(N-1)!}$$

we have that

$$\sum_{k=n}^{N-1} \prod_{m=1}^k \frac{1}{(N-m)\tau} = \frac{1}{(N-1)!} \sum_{k=n}^{N-1} \frac{(N-k-1)!}{\tau^k}$$

Let $j = N - k - 1$, then $0 \leq j \leq N - n - 1$ so that a change of variable results in

$$\sum_{k=n}^{N-1} \frac{(N-k-1)!}{\tau^k} = \frac{1}{\tau^{N-1}} \sum_{j=0}^{N-n-1} j! \tau^j$$

Combining all yields (5), it is

$$\mu_n = \frac{\sum_{j=0}^{N-n-1} j! \tau^j}{\sum_{j=0}^{N-1} j! \tau^j} = \frac{p_{N-n-1}(\tau)}{p_{N-1}(\tau)}$$

which is a fraction of two polynomials of the type $p_m(z) = \sum_{j=0}^m j! z^j = 1 + z + 2!z^2 + \dots + m!z^m$ with positive coefficients (all derivatives are positive). Thus, $p_m(z)$ is rapidly increasing for $z > 0$ and possible real zeros are negative.

The ratio $\frac{j!z^j}{(j-1)!z^{j-1}} = jz$ of two consecutive terms in the polynomial $p_m(z)$ indicates that, if $jz > 1$ holds for all $1 \leq j \leq m$, the terms are increasing, while if $jz < 1$ for all j , the terms are decreasing. Hence, if $j\tau < 1$ for all $1 \leq j \leq N - 1$, which is satisfied if $\tau < \frac{1}{N-1}$, then the terms in $p_{N-1}(\tau)$ as well as in $p_{N-n-1}(\tau)$ are decreasing and both $p_{N-1}(\tau)$ and $p_{N-n-1}(\tau)$ tend to each other so that $\mu_n \rightarrow 1$. In the other case, for $\tau > \frac{1}{N-1}$ and for sufficiently large N , the polynomial $p_{N-1}(z)$ will be dominated by the largest term and μ_n is approximately equal to

$$\begin{aligned} \mu_n &\approx \frac{(N-n-1)! \tau^{N-n-1}}{(N-1)! \tau^{N-1}} = \frac{(N-n-1)!}{(N-1)!} \frac{1}{\tau^n} = \frac{1}{(N-1)(N-2)\dots(N-n)} \frac{1}{\tau^n} \\ &= \frac{1}{((N-1)\tau)^n \left(1 - \frac{1}{N-1}\right) \left(1 - \frac{2}{N-1}\right) \dots \left(1 - \frac{n-1}{N-1}\right)} \end{aligned}$$

If $n \ll N$, then we arrive at formula (6)

$$\mu_n \approx \frac{1}{((N-1)\tau)^n}$$

because $x = \frac{\tau}{\tau_c^{(1)}} = \lambda_1 \tau = (N-1)\tau$ for the complete graph K_N as $\lambda_1(K_N) = N-1$.

References

- [1] van de Bovenkamp, R.: Epidemic processes on complex networks: modelling, simulation and algorithms. Ph.D. thesis, Delft University of Technology, The Netherlands (2015)
- [2] van de Bovenkamp, R., Van Mieghem, P.: Survival time of the Susceptible-Infected-Susceptible infection process on a graph. *Physical Review E* **92**(3), 032,806 (2015). DOI 10.1103/PhysRevE.92.032806
- [3] Cator, E., Van Mieghem, P.: Susceptible-Infected-Susceptible epidemics on the complete graph and the star graph: exact analysis. *Physical Review E* **87**(1), 012,811 (2013). DOI 10.1103/PhysRevE.87.012811
- [4] Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry* **81**(25), 2340–2361 (1977). DOI 10.1021/j100540a008
- [5] Khanafer, A., Baar, T., Gharesifard, B.: Stability properties of infected networks with low curing rates. In: 2014 American Control Conference, pp. 3579–3584 (2014). DOI 10.1109/ACC.2014.6859418
- [6] Li, C., van de Bovenkamp, R., Van Mieghem, P.: Susceptible-infected-susceptible model: A comparison of N -intertwined and heterogeneous mean-field approximations. *Physical Review E* **86**(2), 026,116 (2012). DOI 10.1103/PhysRevE.86.026116
- [7] Pastor-Satorras, R., Castellano, C., Van Mieghem, P., Vespignani, A.: Epidemic processes in complex networks. *Reviews of Modern Physics* **87**(3), 925–979 (2015). DOI 10.1103/RevModPhys.87.925
- [8] Sander, R.S., Costa, G.S., Ferreira, S.C.: Sampling methods for the quasistationary regime of epidemic processes on regular and complex networks. arXiv:1606.00036 (2016)
- [9] Van Mieghem, P.: Decay towards the overall-healthy state in SIS epidemics on networks. arXiv:1310.3980 (2013). ArXiv: 1310.3980
- [10] Van Mieghem, P.: Performance analysis of complex networks and systems. Cambridge University Press, Cambridge (2014)
- [11] Van Mieghem, P.: Approximate formula and bounds for the time-varying susceptible-infected-susceptible prevalence in networks. *Physical Review E* **93**(5), 052,312 (2016). DOI 10.1103/PhysRevE.93.052312
- [12] Van Mieghem, P., van de Bovenkamp, R.: Accuracy criterion for the mean-field approximation in susceptible-infected-susceptible epidemics on networks. *Physical Review E* **91**(3), 032,812 (2015). DOI 10.1103/PhysRevE.91.032812
- [13] Van Mieghem, P., Omic, J., Kooij, R.: Virus Spread in Networks. *IEEE/ACM Transactions on Networking* **17**(1), 1–14 (2009). DOI 10.1109/TNET.2008.925623
- [14] Wang, Y., Chakrabarti, D., Wang, C., Faloutsos, C.: Epidemic spreading in real networks: an eigenvalue viewpoint. In: 22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings, pp. 25–34 (2003). DOI 10.1109/RELDIS.2003.1238052

Part VI
Resilience and Control

Robustness of Network Controllability to Degree-Based Edge Attacks

Jijju Thomas, Supratim Ghosh, Deven Parek, Derek Ruths and Justin Ruths

Abstract We analyze the tolerance of network controllability to degree-based edge attacks as well as random edge failure. In particular, we leverage both control-based and reachability-based robustness metrics to investigate the case when a fixed number of controls are allowed to change locations following each attack. This ability to change the locations of controls models the more realistic scenario in which operators may have a fixed budget of resources but that these resources can be redeployed in response to attacks on the system. We also identify that the most potent targeted attack for network controllability selects edges (on average) based on betweenness centrality.

1 Introduction

Due to their ubiquitous appearance in many applications, such as economics, transportation, biochemical processes, and power systems, large-scale complex networks have received widespread attention from the control community in recent years [10, 19]. Classical control techniques, however, perform poorly for these network structures since they do not scale well with size and complexity. Moreover, in many situations the exact system parameters (the strengths of the interconnections between nodes) are not known. Thus, structural control tools are used to analyze and design various properties of large-scale networks, such as controllability.

Jijju Thomas (e-mail: jijjuthomas@gmail.com) · Supratim Ghosh (e-mail: supratim_ghosh@sutd.edu.sg)
Singapore University of Technology and Design

Deven Parekh (e-mail: deven.svnit@gmail.com) · Derek Ruths (e-mail: druths@networkdynamics.org)
McGill University

Justin Ruths (e-mail: jruths@utdallas.edu)
University of Texas at Dallas

Supported by the SUTD-MIT International Design Centre (Grant IDG31300103)

Although a large body of work addresses static networks, one of the defining characteristics of real-world networks is their natural processes of growth and change [1]. This is true for social networks in which friendships come and go, or also in engineered networks, such as power distribution networks, where natural events can interrupt service due to power line damage. More recently, concerns over security of cyber-physical systems has drawn more attention to targeted attacks with malicious intent. Analysis of network controllability under such attacks not only helps to identify the most vulnerable points in the network but also informs about the robustness of the communication structure. Such information can be used to design networked systems to make them resilient to failures and attacks. Several recent studies have begun to address the robustness of network controllability under various types of attacks [11, 14, 15, 18].

Most of this recent work studies the increase in the number of controls required to recover network controllability following failures of links or nodes [14, 18]. This kind of control-based robustness analysis assumes that the network operator has the capability to add additional controls at any location in the network. A more realistic assumption is that managers have a fixed budget or have a limited quantity of resources that can be deployed in response to an attack or failure. Moreover, the increase in additional controls is only a proxy for the most relevant information - how much of the network is still controllable.

In response to these ideas, [15] introduces a new type of reachability-based robustness metric which captures the extent to which a network remains controllable in the face of an attack. While this allows us to directly study the change in number of controllable nodes, the work required the location of the controls to stay fixed throughout the process. In this paper, we assume that the designer has the ability to relocate the same fixed number of controls after an attack occurs. We quantify the advantage gained by allowing the control input locations to be *free* rather than *fixed*.

The first contribution of this paper is to frame the free- versus fixed-controls scenarios and provide graphical algorithms that allow us to analyze these robustness schemes. Subsequently, we characterize the behavior of synthetic networks (both random Erdos-Renyi networks and scale-free Barabasi-Albert networks) under sequential degree-based edge attacks or random edge failure. We study the robustness of these networks based on three metrics: increase in the number of controls to achieve complete controllability (control-based robustness), and decrease in the number of controllable nodes under fixed and free controls (reachability-based robustness). In doing so we identify the most damaging form of degree-based edge attack and demonstrate that potency of attacks is directly correlated to the betweenness centrality of the removed edges. While it is known in literature that scale-free networks are highly robust to failure but sensitive to targeted attack when it comes to connectivity [1, 4], we establish that in the context of controllability both scale-free and random networks behave in a consistent manner; i.e., the order of potency of attack types are identical for both network models.

2 Background

We consider networks whose state evolves according to linear time-invariant dynamics,

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad (1)$$

where $\mathbf{x}(t) \in \mathbb{R}^N$ and $\mathbf{u}(t) \in \mathbb{R}^m$ represent the state and externally applied input. The matrices $\mathbf{A} \in \mathbb{R}^{N \times N}$ and $\mathbf{B} \in \mathbb{R}^{N \times m}$ denote the state and input matrices, respectively. The system (1) is controllable if and only if the rank of the controllability matrix $\mathbf{C} = [\mathbf{B} \ \mathbf{A}\mathbf{B} \ \cdots \ \mathbf{A}^{n-1}\mathbf{B}]$ equals N [3].

To make the controllability analysis of such systems feasible for large-scale networks, we focus on structured linear systems, where the parameters (entries) of system matrices \mathbf{A} and \mathbf{B} are unknown, but their sparsity structures are known. Such matrices are called structured matrices, where the entries are either fixed zeros or free independent parameters. Systems with structured matrices are said to be controllable if there exists a non-structured real-valued system with the same sparsity structure that is controllable in the classical sense [9]. The control properties of structured systems are defined in a generic sense; i.e., they hold for almost all choices of parameters except for a set of Lebesgue measure zero [9, 20].

Directed graphs offer an attractive alternate approach to study linear systems. The linear system (1) can be represented by a directed graph $G(\mathbf{A}, \mathbf{B}) = (V, E)$ where $V = \{x_1, \dots, x_N, u_1, \dots, u_m\}$ denotes the set of state and input nodes and $E \subseteq V \times V$ denotes the set of edges. We say that an edge exists from $x_j \in V$ to $x_i \in V$ ($u_j \in V$ to $x_i \in V$) if and only if the (i, j) th entry of the matrix \mathbf{A} (the matrix \mathbf{B}) is not a fixed zero. Standard notions of paths, cycles, walks, and families of paths and cycles related to directed graphs are used throughout the paper. However, we also use the somewhat less standard graphical notion of *cactus* for studying the properties of structured linear systems. A cactus is a subgraph of $G(\mathbf{A}, \mathbf{B})$ that consists of a stem (a directed path) with buds (cycles) connected from the stem or from other buds via *distinguished edges*.

It is well known that the existence of a cacti structure (i.e., a single cactus or multiple disjoint cacti) originating from the input nodes (i.e., $\{u_1, \dots, u_m\}$) spanning all the state vertices is both necessary and sufficient for the overall system to be structurally controllable [6, 9]. A cacti structure is the minimum graphical structure that satisfies two fundamental conditions which make the system controllable: all state nodes must be reached from at least one input node; and there must be a sufficient number of inputs to properly control each of the nodes [5]. Loss of either one of these properties renders the system uncontrollable.

A maximum matching algorithm, which runs in polynomial time, can be used to determine the spanning cacti structure that guarantees structural controllability with the fewest number of input nodes required [5, 12]. The algorithm produces the set of edges that compose the stem(s) and the cycle(s) of the cacti. Joining together these edges into stems and cycles and identifying distinguished edges constructs the overall cacti structure. The matrix \mathbf{B} is constructed by connecting each of the base nodes of the stem(s) from an input node (if for some cycles no distinguished edge exists in the state connectivity given by \mathbf{A} , then a distinguished edge must be

added directly from one of the input nodes, i.e., added to \mathbf{B}). Due to the degeneracy of the maximum matching (there are possibly multiple matchings with equal number of matched edges), the spanning cacti structure is, in general, not unique; however, the minimum number of input nodes required is unique and is a feature of the state connectivity matrix \mathbf{A} .

2.1 Robustness Metrics for Network Controllability

In this study we are interested in capturing the ability of the network to retain controllability in the face of edge removal, i.e., changes in topology. We, and the related literature, therefore, use a metric that quantifies the change in network controllability over the course of these changes. In this work, as a continuation of our prior work in [15], we advance the notion of reachability-based robustness as a more direct measure of robustness of controllability and compare that with control-based robustness.

Control-based robustness metric. This is the standard metric that has been used in literature analyzing network controllability robustness [14, 18]. The metric (N_c) specifies the minimum number of additional controls required to maintain complete controllability of the network after an attack. This number can be determined from the result of a maximum matching run on the attacked graph structure, using the method described above. To understand network controllability from a control-based approach, we observe the increase in N_c associated with a combination of network topology and strategy of edge attack. We will denote the minimum number of controls needed to control the original graph, prior to edge removal, by N_c^0 .

Reachability-based robustness metrics. In comparison to the control-based, the reachability-based robustness metric provides a measure of the number of nodes that are controllable by the chosen inputs after an attack on the network [15]. In the simplest case, we considered a set of *fixed controls*, fixed in terms of both the number of controls and the connectivity of controls to nodes. The fixed control set is given by the set of original minimal controls found by the maximum matching; therefore, the fixed number of controls is given as N_c^0 . The number of nodes that can be controlled (the generic dimension of the reachable subspace) in this fixed controls case is denoted N_f . This value can be found using a weighted maximum matching procedure (described in Section 3) [15].

In this current work, we consider a new model for control in the context of failure and attacks. We seek to understand the value of being able to change the connectivity of the controls to nodes, subject to a fixed number of controls (N_c^0). This situation emulates an operator's ability to redeploy controls after each attack, subject to a fixed budget of controls. The number of controllable nodes in this *free controls* case is denoted N_f . We adapt the weighted maximum matching method from the fixed control case to compute N_f . By definition, we expect that $N_f \geq N_f$.

3 Algorithm

The reachability-based metrics computing N_r and N_f involve identification of the number of nodes which remain controllable after removal of edges under external attacks using a set of a given number of controls, but with fixed or free connectivity to the state nodes. Equivalently, this amounts to the identification of the generic dimension of the controllable subspace for the system after the attack [8, 15]. The solution of this problem has been shown to correspond to finding a cycle partition of $G(\mathbf{A}, \mathbf{B})$ and formulated as an integer linear program [16]. In prior work, we extracted this result and presented it in the context of robustness using the equivalent graphical formulation of a weighted maximum matching, specifically for the case where the locations of control inputs are considered fixed [15]. Here, we generalize the algorithm to compute the value of N_f ; i.e., the situation in which the designer has the flexibility to change the locations of controls after the attacks. For the sake of completeness, we first describe in brief the original procedure to find N_r , followed by the modifications necessary to extend it to the case of free controls, N_f .

To compute the cacti control structure subject to a fixed set of controls (the structure of \mathbf{B} is known), we form a special weighted bipartite representation G_B of $G(\mathbf{A}, \mathbf{B})$:

1. Remove nodes that cannot be reached by any control.

For all $i, j = 1, \dots, N$ and $k = 1, \dots, m$:

2. Split the remaining nodes into a pair of positive and negative nodes: $x_i \rightarrow x_i^+, x_i^-$.
3. Add unit-weight edges (x_i^+, x_j^-) if $(x_i, x_j) \in E$.
4. Add unit-weight edges (u_k^+, x_j^-) if $(u_k, x_j) \in E$.
5. Add zero-weight edges (x_i^+, x_i^-) (self-loops).
6. Add zero-weight edges (u_k^+, u_k^-) (self-loops).
7. Add zero-weight edges (x_i^+, u_k^-) .
8. Add a weight $W \geq |E|$ to all edges in G_B .

The original unit-weight edges correspond to the existing edges in the network or to edges that connect the inputs to state nodes. Adding the zero weight edges ensures that a perfect matching is possible (to construct the cycle family that covers all nodes). In particular, the cycles in the cycle family must close the path either through a zero-weight edge to a control (edges in step 7) or as a self edge (edges in steps 5 and 6). Note that the control self-loops in step 6 are rarely matched, unless the control nodes are disconnected. Finally the large weight W ensures that the weighted maximum matching finds the perfect matching, i.e., without the extra weight W there may be a collection of true edges (each with weight $W + 1$) that would form a heavier matching without being a perfect matching.

A weighted maximum matching on the bipartite graph G_B yields a set of matched edges. Mapping these edges (only keeping the edges with weight $W + 1$) back into edges in $G(\mathbf{A}, \mathbf{B})$ constructs a set of paths and cycles. Following the same procedure described for the paths and cycles found by a maximum matching (identifying distinguished edges), the cacti can be formed. The nodes contained within the stems

and cycles rooted in a control node are controllable, whereas those in stems and cycles without connection to a control node are not controllable.

If the number of controls is fixed, but the connectivity is flexible, we modify this method slightly to find the location of the connectivity of these free controls and the number of controllable nodes. We first preprocess the graph by adding the fixed number of control nodes and then placing edges from each control node to every state node in the network (equivalent to setting \mathbf{B} to have no fixed zeros). Then we execute the above process for the preprocessed graph $G(\mathbf{A}, \mathbf{B})$. In the process of the weighted maximum matching described above, only one edge from each control can be matched. Thus the matched edges of the form (u_i^+, x_j^-) indicate a connection of inputs to states that yields the maximum number of controllable nodes; the rest of the control connections are removed.

4 Experimental Setup

We now employ these developed methods to empirically calculate the evolution of the number of controls, N_c and the number of controllable nodes, N_r (fixed controls) and N_f (free controls), as the topology of the network changes due to random edge failure and targeted edge attack. We remove 90% of the total number of edges (L) in the network in steps of 5% (if $0.05L$ is not an integer number of edges, we round down). We denote the number of edges removed as ℓ . This procedure is repeated for 100 different networks (with the same network type, attack type, and average degree) and the network statistics and robustness metrics are averaged over these 100 networks.

While there are a number of interesting features to be studied, here we, in particular, aim to (1) determine the potency of various degree-based attacks on network controllability, and (2) quantify the advantage of being able to reconfigure controls following an attack.

Random network models provide an efficient platform for studying change in network properties because they can be synthesized according to strict criteria, such as a fixed average degree. We use Erdos-Renyi (ER) and Barabasi-Albert (BA) models generated with $N = 1000$ nodes and average degrees $q = 2, 4, 6$. These models represent the stereotypes of random and scale-free networks and are commonly used to represent these fundamental classes of network topology. The Erdos-Renyi model is used for generating random graphs by connecting nodes randomly with probability p [7]. The Barabasi-Albert model starts with a clique of q nodes. A node is added to the graph and q edges establish connection from the new node to the nodes already in the network, preferentially biased towards nodes with high degree. This process is repeated until the graph has n nodes [2].

The networks generated using the BA and ER models were subjected to random edge failures as well as targeted edge attacks, and the change in network controllability was analyzed using both control-based and reachability-based robustness metrics. A wide variety of edge attacks are viable, however, we restrict our attention to degree-based attack strategies because degree is a local property of the network.

More sophisticated attacks may be possible - for example based on the paths and cycles of the cacti - however, they would require global information about the network, which is less likely in terms of the capabilities of an attacker. There is precedent that connects node and edge degrees to controllability [17].

- *Random edge attack* emulates a spontaneous failure of a connection by selecting an edge uniformly randomly.
- *Degree-based edge attack* removes edges in decreasing order (largest first) in terms of the in-, out-, or total-degree characteristics of the edge's source and target nodes. In our terminology the attack is denoted, for example, *in-out* if the edge is ranked highly for having a high combined in-degree of the source node and out-degree of the target node. We consider all combinations: *in-out*, *in-in*, *out-in*, *out-out*, as well as an attack simply based on the *total* degree of the source and target node.
- *Betweenness centrality edge attack* is not a degree-based attack, however, we will show in Section 5 that this attack provides insight into the potency of the degree-based attacks. The betweenness centrality of an edge is related to the number of shortest paths passing through that edge [13].

5 Results

For all network types and average degrees (as a sample, Fig. 1 presents ER networks with average degree $q = 2$), the in-out attack is decisively the most potent attack in terms of loss of network controllability; out-in is definitively the weakest attack. The remaining degree-based attacks fall in between these extremes and are not significantly different from random failure. This trend is so strong that we have dropped, for visual clarity, all other attacks in the presentation of Figure 3, which shows the evolution of the reachability-based robustness measures N_r (fixed controls) and N_f (free controls), as well as the control-based robustness measure N_c under different attacks types on BA and ER networks with average degree $q = 2$ and $q = 6$. We use $n_\alpha = N_\alpha/N$ to denote the fraction with respect to the total number of nodes, for $\alpha \in \{c, r, f\}$.

The literature on network robustness - as opposed to this work on network *controllability* robustness - studies how the connectivity (e.g., the diameter) of the network changes in response to edge failure and attack [1]. A fundamental result in this body of work is that scale-free networks like BA tend to be highly robust to random failure, but highly sensitive to degree-based attacks. In contrast random graphs like ER evidence a more moderated response to both failure and attack. Our results here reveal that this differentiation between ER and BA networks does not exist in the context of robustness of network controllability.

A few other observations help us to understand the potency of the in-out attack. Fig. 1 also plots the evolution of the number of strongly connected components (SCCs) as edges are removed. A strongly connected component \mathcal{S} is a maximal subgraph such that every node $u \in \mathcal{S}$ can reach every other node $v \in \mathcal{S}$ along a

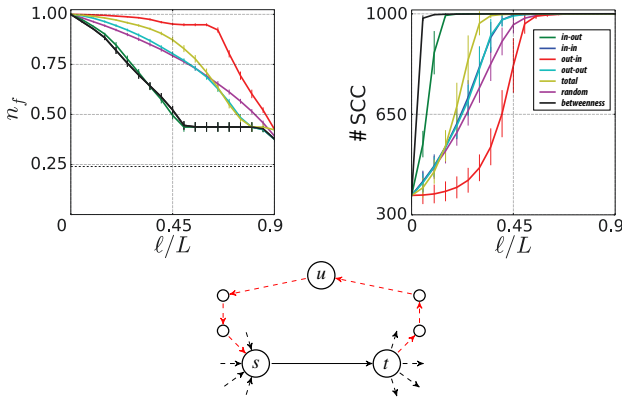


Fig. 1: Fraction of controllable nodes in the free controls case and the saturation of strongly connected components; all attack types; ER networks with $q = 2$. A schematic showing that edges with high in-out degree have a greater likelihood of belonging to an SCC.

directed path. The smallest SCC possible is, therefore, a single node. In ER graphs, we observe a saturation of the number of SCCs (see Fig. 1), effectively witnessing the breakdown of larger SCCs into smaller ones until all SCCs are single nodes (because there are 1000 nodes). Because BA networks are acyclic, the number of SCCs is always 1000 (we omit these figures). The rate and early onset of the saturation of SCCs corresponds directly to the potency of the attack.

While the precise causal link between SCCs and network controllability is a topic for future work, our observations do, however, help to lead us to an explanation behind the potency of the in-out attack, and, therefore, degree-based attacks in general. Consider an edge with high in-out degree, as shown in the schematic in Fig. 1. We argue that the high out-degree of the target node and the high in-degree of the source node increase the likelihood for there to be a path from the target node back to the source node. This return path would imply that the source and target nodes are part of an SCC; and most importantly, the edge in question is part of an SCC. We have now argued that the in-out attack is likely to remove edges within SCCs, but have not yet made the connection that relates this back to network controllability.

The high in-degree of the edge’s source node and high out-degree of the edge’s target node makes the edge a natural bridge between parts of the network (Fig. 2). This notion of edges being “bridges” between nodes is similar to the concept of *betweenness centrality*, which ranks edges according to the number of shortest paths that pass through an edge. More precisely, between all pairs of nodes in a network there exists a minimum number of edges that separates them, although there could be several such shortest paths that achieve this minimum. Betweenness centrality of an edge is the fraction of shortest paths between two nodes containing the edge,

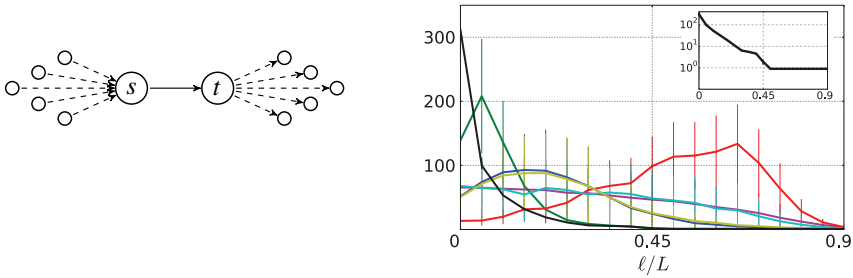


Fig. 2: Average betweenness centrality of the edges removed for each attack step in BA networks with $q = 6$. The schematic shows that edges with high in-out degree are more likely to act as bridges between parts of the network.

summed over all possible pairs of nodes in the network. Edges that connect clusters of nodes act as bridges between these nodes and, therefore, will be involved in many of the shortest paths between nodes in the different clusters.

We assert that the ranking of an edge according to in-out degree correlates strongly with a ranking by betweenness centrality. Therefore, we ascribe the potency of the in-out attack to the fact that it is a rough proxy for betweenness centrality. Fig. 2 shows the average betweenness centrality of the edges removed at every attack step, for each attack type, revealing that the in-out attack (green curve) indeed targets high betweenness edges much more directly than any other attack.

To cement the connection relating betweenness centrality to the potency of the degree-based attacks, we define a new non-degree-based attack targeting edges in decreasing order of their betweenness centrality. It is evident from Fig. 3 that this attack matches or outperforms the in-out attack, and all other attacks, (while $0 \leq \ell/L \leq 40\%$ for BA $q = 6$). For low average degree, the in-out attack tends to be a more accurate proxy for betweenness centrality. The reason why in-out attack marginally outperforms the betweenness centrality attack (past $\ell/L > 40\%$ for BA with $q = 6$) is due to the fact that once the network connectivity is sufficiently sparse, the betweenness centrality of all remaining edges is 1 (i.e., there is mainly just one shortest path between pairs of nodes). Therefore, beyond this point the betweenness centrality attack is no longer able to be discriminatory between edges, whereas the in-out attack still provides a meaningful ranking based on the local degree information. We can see the betweenness centrality flatten out at 1 in the inset plot in Fig. 2. This point occurs when the potency of the in-out attack overtakes the betweenness centrality attack - around 40% for BA $q = 6$.

In summary, we observed that the in-out attack seems to aggressively destroy SCCs if they are present. This motivated the notion of viewing high in-out edges as bridges, which connects with betweenness centrality. Through a direct evaluation of the betweenness centrality of the edges removed under different degree-based attack schemes and by implementing a betweenness centrality based attack, we clearly connect betweenness centrality with the most rapid decrease in network controllability. Although there are possibly other even more potent attacks, they

would require global knowledge of the network topology. We use the betweenness centrality attack not as a focus of this paper, but as a benchmark to explain the behavior of the in-out degree based attack.

These same arguments can also be used to explain the performance of the other degree-based attacks as well. In particular, edges with high out-in degree function entirely opposite to edges with high in-out degree. Reversing the arrows in the schematics of Figs. 1 and 2 shows that such edges do not participate with high likelihood in SCCs or as bridges. The out-in attack line (red) in Fig. 2 clearly indicates that it systematically selects edges with the lowest betweenness. The peak towards the end is simply the artificial inflation of edges' betweenness due to the removal of all the edges with less betweenness centrality early on in the process. The other attacks tend to fall in between these extremes.

5.1 Free Controls vs Fixed Controls

By construction the number of controllable nodes in the free control case is the same or greater than the fixed control case; here we quantify this improvement. Figure 3 presents a comprehensive picture of all network controllability statistics for BA and ER networks and confirms that $N_f \geq N_r$ for both networks at all average degrees.

The last row of plots in Fig. 3 displays the difference $n_f - n_r$. We observe that the general qualitative curve of these plots is equivalent across network types and average degrees, which emphasizes that free controls provide a systematic benefit over fixed controls, e.g., the initial peak at around 20% of edges removed. This benefit scales based on network type and average degree. A clearer signature difference between fixed and free controls is presented in Fig. 4, where we plot the difference in the changes in reachability- and control-based robustness. More precisely, $\Delta N_\alpha(\ell) = |N_\alpha(\ell) - N_\alpha(0)|$, i.e., the absolute change from the initial value of the robustness metric, for $\alpha \in \{c, r, f\}$. We know that the number of controls, N_c will increase as edges are removed - in particular, for each edge we remove either no new controls will need to be added, or one new control will need to be added to maintain controllability. Therefore, there is a limit on the rate of increase of ΔN_c . On the other hand, ΔN_r and ΔN_f have no such limits on the reduction in controllable nodes; for each edge that is removed, it is possible to lose controllability to many nodes. The quantities $\Delta N_r - \Delta N_c$ and $\Delta N_f - \Delta N_c$, thus capture the extent to which the rate of increase in the number of controls is matched or exceeded by the decrease in the number of controllable nodes under the fixed or free control scheme. One interpretation of Fig. 4 is that it shows that the free control case is able to reduce the loss of controllable nodes to the same rate as the increase in number of controls when the plot has zero value. For the weaker attacks, the ability to move controls is able to compensate for up to about $\ell/L = 0.7$. For the more powerful in-out and betweenness attacks, this is true only up to about $\ell/L = 0.1$. When we compare this to the fixed control case in Fig. 4, we observe that fixed controls are not able to perfectly compensate for any amount of edge loss.

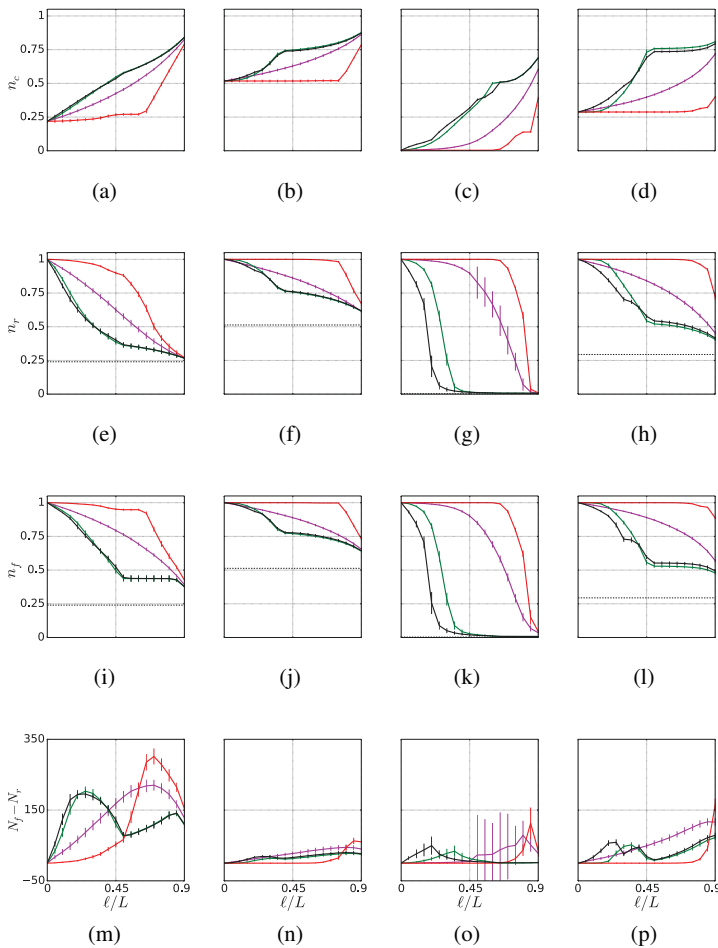


Fig. 3: Change in robustness measures of ER (columns 1 and 3) and BA (columns 2 and 4) networks for $q = 2$ (columns 1 and 2) and $q = 6$ (columns 3 and 4) under in-out (green), out-in (red), random (purple), and betweenness centrality (black) attacks. The horizontal dashed lines in rows 2 and 3 indicate the initial number of required controls, N_c^0 .

6 Conclusion

In this work, we have quantified and analyzed the changes in the controllability of synthetic networks (random ER and scale-free BA networks) in response to degree-based edge attacks using both control- and reachability-based metrics. We identified that the potency a degree-based attack is directly related (on average) to the betweenness centrality of the edges being removed. Moreover, we have discovered that for robustness of network controllability, both random networks models behave in a

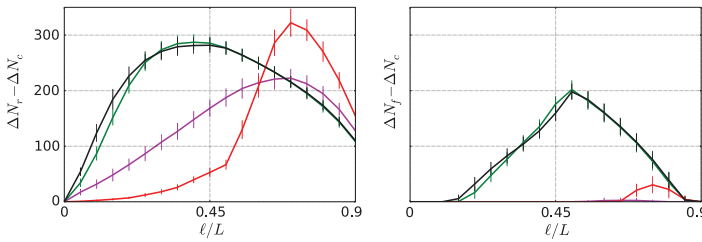


Fig. 4: Cumulative difference between the reachability-based and control-based robustness metrics for the free and fixed cases; ER network with $q = 2$. $\Delta N_\alpha(\ell) = |N_\alpha(\ell) - N_\alpha(0)|$ represents the absolute change from the initial value of the robustness metric, for $x \in \{c, r, f\}$.

very similar manner, contrasting with findings on robustness of network connectivity, where scale-free networks evidence higher robustness to random failures.

References

- [1] Albert, R., Jeong, H., Barabasi, A.: Error and attack tolerance of complex networks. *Nature* **406**(6794), 378–382 (2000)
- [2] Barabási, A.L., Albert, R.: Emergence of scaling in random networks (1999)
- [3] Brockett, R.: Finite dimensional linear systems. Series in decision and control. Wiley (1970)
- [4] Callaway, D.S., Newman, M.E.J., Strogatz, S.H., Watts, D.J.: Network robustness and fragility: percolation on random graphs. *Physical Review Letters* **85**(25), 5468–5471 (2000)
- [5] Commault, C., Dion, J.M., Van der Woude, J.W.: Characterization of generic properties of linear structured systems for efficient computations. *Kybernetika* **38**(5), 503–520 (2002)
- [6] Dion, J.M., Commault, C., van der Woude, J.: Generic properties and control of linear structured systems: a survey. *Automatica* **39**(7), 1125–1144 (2003)
- [7] Erdos, P., Renyi, A.: On random graphs i. *Publicationes Mathematicae (Debrecen)* **6**, 290–297 (1959)
- [8] Hosoe, S.: Determination of the generic dimensions of controllable subspaces and its applications. *IEEE Transactions on Automatic Control* **25**(6), 1192–1196 (1980)
- [9] Lin, C.T.: Structural controllability. *IEEE Transactions on Automatic Control* **AC-19**(3), 201–208 (1974)
- [10] Liu, Y.Y., Slotine, J.J., Barabási, A.L.: Controllability of Complex Networks. *Nature* **473**(7346), 167–173 (2011)
- [11] Mengiste, S., Arvind, A., Kumar, A.: Effect of edge pruning on structural controllability and observability of complex networks. *Scientific Reports* **5**(18145), 4420–4425 (2015)
- [12] Murota, K., Poljak, S.: Note on a Graph-Theoretic Criterion for Structural Output Controllability. *IEEE Transactions on Automatic Control* **35**(8), 939–942 (1990)
- [13] Newman, M.: Networks: an introduction. OUP Oxford (2010)
- [14] Nie, S., Wang, X., Zhang, H., Li, Q., Wang, B.: Robustness of controllability for networks based on edge-attack. *Plos One* **9**(2), e89,066 (2014)
- [15] Parekh, D., Ruths, D., Ruths, J.: Reachability-based robustness of network controllability under node and edge attacks. In: *Signal-Image Technology and Internet-Based Systems (SITIS), 2014 Tenth International Conference on*, pp. 424–431 (2014). DOI 10.1109/SITIS.2014.100
- [16] Poljak, S.: On the generic dimension of controllable subspaces. *IEEE Transactions on Automatic Control* **35**(3), 367–369 (1990)

- [17] Pósfai, M., Liu, Y., Slotine, J.E., Barabási, A.: Effect of correlations on network controllability. *Scientific Reports* **3** (2013)
- [18] Pu, C.L., Pei, W.J., Michaelson, A.: Robustness analysis of network controllability. *Physica A: Statistical Mechanics and its Applications* **391**(18), 4420–4425 (2012)
- [19] Ruths, J., Ruths, D.: Control Profiles of Complex Networks. *Science* **343**(6177) (2014)
- [20] Shields, R.W., Pearson, J.B.: Structural controllability on multiinput liner systems. *IEEE Transactions on Automatic Control* **AC-21**(2), 203–212 (1976)


Use of Random Topics as Practical Control Signals in a Social Network Model

Francesca Casamassima and Marco Cremonini

Abstract In this paper, we study practical strategies for controlling the behaviour of a synthetic social network modelling the dynamic diffusion of knowledge. The problem of controlling the evolution of complex networks has been extensively studied in recent years and remarkable theoretical results have been achieved. However, still largely unexplored is the analysis of realistic control strategies for complex networks and the special case of social networks. Our model of knowledge diffusion in a social network is used for simulating and evaluating possible control strategies of social network behaviour. Our approach is to exploit the controlled injection of random topics into some driver nodes for influencing the overall dynamics. This way, it is possible to modify some key control parameters in a deterministic way with realistic inputs, considering the strong practical constraints of social networks with respect to control measures. Control parameters considered are: The *injection interval* of random topics, the *rate of driver nodes* with respect to the network size, and the *selection criteria* of driver nodes. Finally, we discuss possible applications and the challenges that social networks pose to the issue of network control.

1 Introduction

The idea behind our research started from a simple consideration: Both with synthetic models and in real social networks, it is well-known and documented that a strong tendency of agents towards *polarization* often emerges. Agents often form tight communities with few, if any, weak ties between them, heterogeneity of traits and characteristics of connected agents tend to disappear with the increase of homophily [12, 17]. At practical level, it was often observed how people on social networks tend to slide into "filter bubbles" [9, 22] - i.e., self-reinforcing social contexts dominated by information homogeneity with few occasions to have contacts

Francesca Casamassima · Marco Cremonini (e-mail: marco.cremonini@unimi.it)
Dept. of Computer Science, University of Milan

with critical analyses, information from unaligned sources or contrasting opinions - or, in knowledge diffusion and networked learning, how knowledge sometimes diffuses unevenly, exhibiting strong polarization - e.g., agents showing strong tendency towards specialisation like when students/recipients of information grow strong interests only in a narrow set of topics disregarding the richness of the full information spectrum [2].

In this work we specifically considered research that have applied results from traditional control theory to complex systems. From those theoretical and experimental results we have derived new control strategies for our synthetic model.

Structural controllability of networked systems [15] has been extensively studied since the end of the past decade, following the growing interest in network science [20]. The property of structural controllability is central in the study of how the dynamic of a complex system can be controlled [14]. In short, from control theory, a dynamic system is said to exhibit structural controllability if, with a suitable selection of inputs, it can be driven from one state to any other states in finite time. Inputs to the system are represented by driver nodes receiving external perturbations. Liu et al. in a seminal paper demonstrated how the problem of determining the minimum set of driver nodes required for structural controllability can be mapped into a maximum matching problem [16]. Some remarkable theoretical results have been recently demonstrated for complex networks [6, 18].

However, demonstrating structural controllability for a complex systems is neither always necessary nor sufficient when realistic scenarios are considered [10, 19]. It is not strictly necessary because often it is not required to be able to drive a system from every initial states to every final states. In most practical situations there is the need to tune the dynamical evolution from one trajectory driving the system towards a negative outcome to another trajectory, possibly unknown but leading to a better outcome. So, in many practical situations, it is not an optimisation problem the one we should solve (i.e., applying the best inputs to the smallest set of driver nodes in order to reach the optimal final state), rather it is a problem of perturbing the system evolution for modifying the *basin of attraction* (i.e., modify the dynamics so that the system that was attracted towards a certain state space becomes attracted towards a different one) [5]. The property of structural controllability is not sufficient because often there are many practical limitations to the type of perturbations that could be injected into the system through driver nodes and also limitations to the accessibility of driver nodes. In many real cases, we are neither free to choose the best inputs nor to observe and manipulate all agents.

These two considerations regarding the limitations of structural controllability are the core motivations for the present research. We consider our synthetic model for knowledge diffusion and study the strategic use of random information to enrich the state of some agents as a possible realistic input to driver nodes. In a social context, different from typical industrial case studies of control theory, it is extremely difficult to provide means to change some agents state parameters or deleting some information and replace with others. We do not have actuators for tuning agent' states, there is no hook, knob or controlling interface to handle. The best tools we have in a social context are external perturbations in the form of new information

injected into the system and the reduction of physical and cultural barriers preventing agents to acquire more information. However, attempting to directly and openly influence individuals with messages and actions explicitly tailored to change their opinions, preferences, or interests can easily backfire, as the long experience with many advertising campaigns that ultimately produced adverse reactions (i.e. reactance) [23] or the many criticisms concerning the lack of ethics in the social experiment run by Facebook [13] have witnessed. Differently, attempts to sustain the circulation of information and to increase serendipitous encounters even through digital interfaces (e.g. browsers) have typically received good acceptance and were perceived as beneficial for the social welfare [21]. Therefore, we assume that ethically, culturally and even practically, a strategy for perturbing the behaviour of a social network in order to modify its basin of attraction based on the injection of random information into the system could be accepted as fair and ethical. On the other side, the goal of a control strategy applied to a social context is often to fix a bad trajectory in system evolution, rather than reaching an exactly defined final state. For example, in a networked learning scenario, the goal could be to increase the average level of knowledge on a set of topics by limiting the tendency of agents to dedicate all their interest on few of them only. Similarly the goal could be to limit the degree of homophily to reduce the formation of secluded communities based on same ideology, tastes, or cultural preferences. There are many examples of perverse dynamics affecting a social network ultimately resulting in negative outcomes like network partitions (or quasi-partitions), severe drop in communication, diffusion of knowledge limited to enclaves, or topological structures further limiting the controllability, the observability, and the communication efficiency [4, 7].

In this work, randomness in agent behavior has been modelled as new topics exogenously inserted in agents' state during a simulation: This event wish to represent the typical "unsought encounter" of serendipity and modifies both an agent's criterion of choice of who to communicate with and how knowledge among agents is shared. A study of the adaptation of the serendipity concept for our social network model has been presented in [8]. Results of this work have been produced through simulations defined as variations of a reference configuration of our synthetic social network. In particular, our goal was to study how the network behavior depends from three key control parameters: The *injection interval* of random topics, the *rate of driver nodes* with respect to the network size, and the *selection criteria* of driver nodes. For each parameter, simulations were replicated for different network sizes to verify how the effects of scale influence control strategies.

2 Original Model

In this section we provide a summary of the characteristics of our original agent-based model, before the extensions we made to test control strategies. A more detailed description and analysis can be found in [2]. The model is inspired by question-answer networks where knowledge is shared from expert agents (with respect to a certain topic) answering questions received from less knowledgeable ones.

We assume a set of agents and a set of *topics* to be given. Each agent has a certain level of *interest* and skill (*quality*) on each topic, both change through interactions with other agents. In more detail, we consider a set of N agents, n_1, n_2, \dots, n_N , each one characterized by a *Personal state* PS_{n_i} (what n_i knows) and a *Friend state* FS_{n_i} (who n_i knows). The *Personal state* has the form $PS_{n_i} = (\bigcup_{j \in T_i} (topic_j, quality_{i,j}, interest_{i,j}))$, where T is the set of topics that the population knows; each agent n_i knows a variable subset of them $T_i \subseteq T$. The *Friend state* has the form $FS_{n_i} = (\bigcup_{j \in N_i} (n_j, answers_{i,j}))$, where n_j are the identifiers of agents connected with n_i and $answers_{i,j}$ is a counter to keep track of the number of interactions with each peer. The setup has been defined to be the most neutral, with topics T_i assigned to each agent and associated qualities selected randomly, interests distributed uniformly and no connection.

A network is dynamically formed according to the following steps:

1. At each tick, an agent $n_{i'}$ is randomly selected with no repetition (i.e., the simulator can only run actions on one node for each tick), then a topic ($topic_{j^*}$) is selected from its Personal state. The choice of the topic is a weighted random selection with values of the associated interests ($interest_{i',j^*}$) as weights, this way topics with higher interest are more likely to be selected;
2. Among $n_{i'}$ "friend" agents and their "best friend" holding topic ($topic_{j^*}$), select agent $n_{i''}$ with maximum value of topic's quality ($quality_{i',j^*}$);
3. If $quality_{i'',j^*} > quality_{i',j^*}$ then the communication takes place and agent $n_{i'}$ increases $quality_{i',j^*}$ of $topic_{j^*}$;
4. Otherwise, if either step 2 or 3 fail (i.e., there is no 1-step or 2-steps connected agent holding $topic_{j^*}$ with a topic's quality greater than that of agent $n_{i'}$) then select an agent $n_{i'''}$ at random among the population;
5. if $n_{i'''}$ holds $topic_{j^*}$ and $quality_{i''',j^*} > quality_{i',j^*}$, then the communication takes place and $quality_{i',j^*}$ increases, otherwise the communication fails.

Best "friend-of-friends". Given agent $n_{i'}$, and a selected $topic_{j^*}$, for each of its friends, the "best friend" agent is the one owning $topic_{j^*}$ and the higher value of the attribute *answer*. The reason for this solution is that we consider unrealistic in a social context to scan all agents with a distance of 2 from the one selected. The selection based on the *answer* attribute represents a basic form of transitive trust. It is worth noting that the inclusion of "best friends" fosters network transitivity and the formation of triads, two key characteristics of social networks.

Start up. At start up, agents have no connection (i.e., Friend state is empty). When, for an agent, the 5-steps algorithm is executed, a topic is selected in *Step1*, then *Step2* and *Step3* fail and in *Step4* a random agent is selected. If *Step5* succeeds, then the connection is established. This mechanism triggers the network formation at start up.

State update: Quality and Interest. After a successful interaction, the agent that started the communication is updated. For model simplicity no change in the respondent's state is produced, because knowledge, being an intangible good, does not decrease when shared, and we assume no cost for the transmission. *Quality* and *interest* are always non negative quantities. For the quality parameter associated to each topic an agent owns, we decided that it simply increases in chunks calculated as

a fraction of the knowledge difference between two interacting agents. This implies that in subsequent interactions between two agents (and assuming the quality of the more expert stay the same), the less expert accumulates knowledge in chunks of diminishing size. For completeness, in a more elaborate version of the model, we assumed the presence of distrust. In that case, an agent distrusts another one when they interact for the first time and the distrust progressively vanishes as successfully interactions occur. Distrust was modelled as a discount rate going to zero exponentially. Motivations for the assumption could be found in the literature about collective behavior [11] and refers both to the prevalence of egocentrism in assimilating new information and to trust dynamics.

The dynamics we have assumed for the *interest* associated to the topic for which the interaction took place is similar to the previous case, but with the difference that the sum of the interests in different topics of an agent is a bounded value. This means that when the interest associated to a certain topic increases, interests associated to the other topics decrease uniformly. Motivations for this assumption can be found in cognitive science studies, which have shown the tendency of people to shift their attention and interest, rather than behave incrementally [11], and in associating the interest for a topic to the time spent dealing with that topic (studying, experimenting, etc.); in this sense the sum of all interests per time period (day, week, etc.) has an upper bound.

For space limitation, we do not present here the functional forms of quality and interest, the metrics defined for measuring the model dynamic behavior, and its analysis. Interested readers could find them discussed in [2]. For the scope of the present work, the following aspects are important to know:

- some agents become hubs, receiving a disproportionate amount of questions;
- a giant component typically emerges in the network;
- agents tend to polarize their interests on only few of the topics owned;

Therefore, in a network produced by our model, agents have clearly different roles with respect to communication, there are very few isolated nodes or components, and the diffusion of knowledge is globally uneven, and locally very skewed on just few topics. In other words, some super-experts emerge in a population of specialised individuals, rather than generalists.

3 Control Strategies

The original model was modified with the aim of permitting to select certain nodes as driver nodes and to modify the local state by inserting some random topics, according to a given frequency of perturbation. Injected random topics wish to represent an external perturbation in the form of information already existing into the system forced to circulate between agents. These are not brand new information inserted into the network, for this reason the mechanism can be also seen as a solution to lower barriers to information spreading. The goal is to study how the behaviour of

the network changes as a result of such a perturbation and how this approach could be used as a control strategy.

With respect to the mechanism that the original model implemented to construct a network, the injection of random topics in selected driver nodes has the effect of triggering new communications among agents. More specifically, with new random topics, the network construction mechanism strictly based on topology is occasionally bypassed and a rewiring effect is produced. In the results we will see that it is this rewiring effect the key for controlling some network characteristics.

The base configuration that we perturbed with random topics typically produces a giant component, some nodes emerge as hubs, and interests are often polarized on few topics per node. The following parameters are fixed for all simulations we run: *Number of information in the system* $|T|$: 100; *Max Number of information per node at setup* λ_T : 10; *Duration of simulation (#ticks)* Γ : 100000. These values have been chosen for presentation sake among the many tested as representative of typical behaviours.

The model configurations representing the control strategies of this study correspond to network setup obtained adjusting one of the following features:

- *Selection of driver nodes*: the top 1% to 30% of nodes ordered based on decreasing *node degree* or *betweenness*;
- *Amount of perturbation*: from 1% to 30% of topics owned by each driver node are added as new random topics;
- *Periodicity of perturbation*: from every 1000 to every 5000 ticks driver nodes are modified with random topics.

For sake of presentation, we only present the results of six configurations (**C1-C6**) defined by parameters showed in Table 1. In addition, to study the effects on network behavior of each control strategy with respect to the *network size*, each configuration is tested with network size increasing from 100 to 1000 nodes. We limited the maximum size to 1000 for practical reasons. In particular, for larger networks the dynamics become very slow due to communication congestions provoked by the diminishing ratio between the number of information in the system ($|T| = 100$) and the number of agents. By testing with longer simulation periods or with more information we did not observe meaningful differences with respect to the results already obtained.

Configuration **C0** serves as a benchmark, representing a typical network behaviour with no injection of random topics in nodes.

With configurations **C1**, **C2**, **C3**, we show how the network dynamics reacts to different number of perturbations. This suggests how frequent should be the control input for achieving a certain effect. For these three configurations, the other key control parameters are: driver nodes are selected as the 10% of higher degree nodes, and the number of random topics injected at each perturbation equals to 30% of one node's topics. The choice of 10% and 30%, again, is mostly for presentation sake. However, we note that these are values aligned with the real ones for network controllability of social communication networks, as showed in Table 1 of [16].

With configuration **C4** and **C5**, we wish to show how the control strategy works with few driver nodes. We present the results for rate of driver nodes, of 1% and 5%, which are actually small values for driver nodes' rate, similar only to those of small intra-organizational networks presented in [16]. For these two configurations, the other control parameters are set to 30% for the random topics injected (perturbation value), and a perturbation interval of 1000 ticks. This way, **C4** and **C5** are actually variations of **C1** for different number of driver nodes.

Finally, with configuration **C6** we discuss an example of driver nodes selected from a list of nodes ordered for decreasing betweenness rather than degree. Again, for facilitate the comparison, the other parameters are the same of configuration **C1**.

Configuration C0	Configuration C1, C2, C3
Selection criteria: none	Selection criteria: node degree (decreasing)
# of driver nodes: 0	# of driver nodes: 10% of nodes
Perturbation value: 0	Perturbation value: 30% of new random topics
Periodicity: 0	Periodicity (ticks): 1000 (C1), 2000 (C2), 5000 (C3)
Configuration C4, C5	Configuration C6
Selection criteria: node degree (decreasing)	Selection criteria: node betweenness (decreasing)
# of driver nodes: 1% (C4), 5% (C5)	# of driver nodes: 10%
Perturbation value: 30%	Perturbation value: 30%
Periodicity: (ticks): 1000	Periodicity (ticks): 1000

Table 1: Control strategies: configuration parameters

4 Simulation Results and Discussion

We first present in Figure 1 the results comparing the behaviours of the six control strategies and the original network with no control. In addition, Figure 1 also compares the same control strategies for two network sizes of N=200 and N=1000 to show how the effects changes on different scale.

Average Node Degree. In general for both network sizes, we observe that injecting driver nodes with random topics produces a sensible reduction of the average degree with all control strategies with respect to the original network. Even configuration **C3**, for which a weaker effect was expected being the one with less perturbation events, produces a remarkable reduction at both network sizes. Qualitatively, this may suggest that injecting random information into driver nodes with high node degree could drive a reduction of a key network parameter as the average node degree. Configuration **C4** presents an interesting case. This is the one selecting the smallest number of driver nodes. For the smaller network (N=200), it just selects 2 driver nodes, likely hubs, and produce a strong reduction effect on the average degree. This result confirms what has been already found theoretically, that few driver nodes are responsible for a large average degree reduction. With the larger network (N=1000),

10 nodes are selected and the effect becomes weaker, similar to **C3**. Another effect we observed is that in these tests, network hubs are certainly selected as driver nodes and while globally control strategies strongly reduce the average node degree, on hubs the effect is the opposite, their degree increases, so they become even more relevant for the network communication.

Clustering Coefficient. Associated to the general reduction of the average degree, the other important effect on a key network parameter is that the clustering coefficient generally increases, with respect to **C0**. This signals that the communication has become more decentralised and local (i.e., more triangles have formed), a direct effect of the the rewiring effect carried by the introduction of random topics. Together, the reduction of the average degree and the increase of the clustering coefficient tends to enhance the small-world characteristics of the network, which is another important effect in terms of controllability.

Connected Components. An effect of increasing the network size is that the formation of a giant component becomes slower and some disconnected components may persist. We can see this effect comparing the Connected Components graphs for the two sizes. With $N=200$ there is always one component, while for $N=1000$ the number varies with some configurations producing more than one component, signalling that the convergence of the network towards a single giant component has become much slower.

Overall, these results are in line with the theoretical results of [16], which has demonstrated that the less heterogeneous in degree is a network, the more is controllable (the fewer the driver nodes). Therefore, with the injection of random topics, as we expected, we increase the theoretical controllability of our network.

Finally, in Figure 2 we compare how the network behaves, represented by average degree and average clustering coefficient, with some control strategies on an extended range of network sizes. The goal here is to better present how the network size may affect the results. Specifically, we compare *Configuration C1*, to *Configuration C4* having a reduced number of driver nodes (1%), and to *Configuration C6* that uses the betweenness as the selection criteria for the agents.

In general, we observe that the number of driver nodes is the critical parameter. If driver nodes are too few the control becomes weak, as in case of **C4**, where it is evident that for $N=100$ the 1% rate of driver nodes, which means selecting just the single node with highest degree, is insufficient to modify the dynamics. **C1** and **C6**, with 10% of driver nodes, perform much better at $N=100$.

However, many driver nodes, thus high level of rewiring, may introduce communication inefficiencies. The same **C4** produces better results (i.e. larger average degree reduction) than **C1** and **C6** from $N=200$ to $N=500$. For networks larger than $N=500$, **C1** and **C6** perform better again. These results present an interesting practical control problem still not fully investigated in the literature. While some theoretical results have been studied for reference network topologies [5, 16, 18], very few has been done in terms of mechanisms for dynamically adjusting the degree of control on a live situation, when the social network is actually evolving.

With respect to configuration **C6** using betweenness instead of node degree for ranking nodes, the results are actually very similar to **C1**. After a more detailed

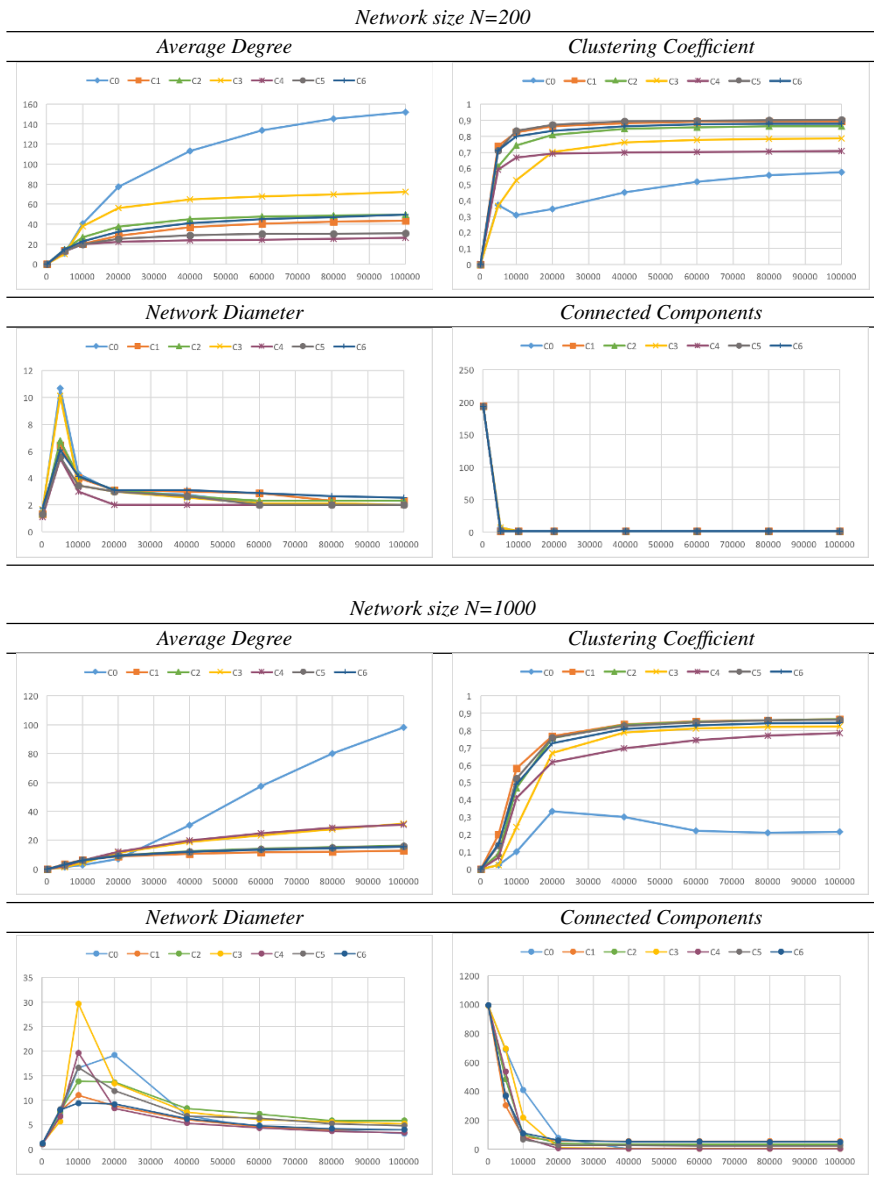


Fig. 1: Results for the different configurations along the simulation time and for two network sizes. x -axes represent the simulation time (ticks), y -axes the absolute values of metrics.

investigation we have found that the reason is because the nodes selected with the betweenness ranking largely overlaps with those selected with the degree ranking. This depends in part on the peculiarities of our model and in part from topology. Clearly, while in this work we started by selecting driver nodes based on node degree and betweenness because of their relevance as network parameters, and because several studies had presented theoretical analyses focused on them, many other possibilities are still unexplored. We plan to consider some of them in future works.

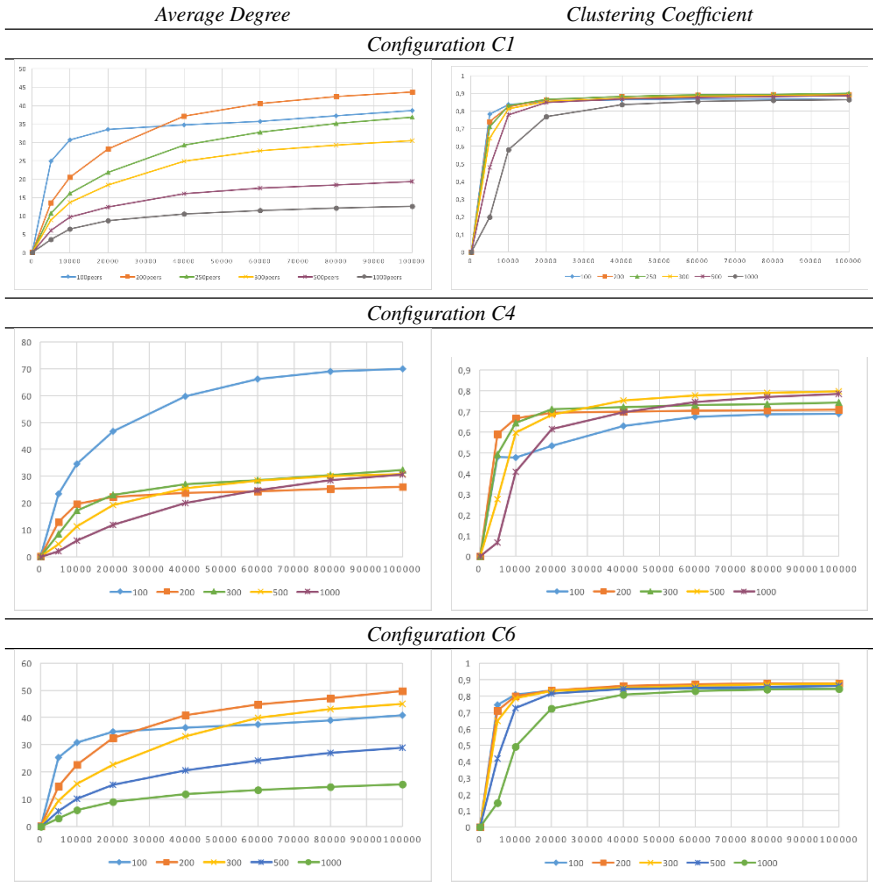


Fig. 2: Average degree and clustering coefficient in case of perturbation every 1000 ticks and 10% of driver nodes (C1), perturbation every 1000 ticks and 1% of driver nodes (C4), and using betweenness instead of the average degree for node selection (C6). For every configuration, the results with different network sizes are presented. x-axes represent the simulation time (ticks), y-axes the absolute values of metrics.

5 Conclusions

In this paper we discussed a possible use of random topics as control inputs for driver nodes of a social network. The idea of using random information in control strategies has some common aspects with research in recommendation systems, which have the same problem of polarization of interest and the need to improve diversity that we consider for knowledge diffusion [1, 3]. However, explicitly considering random information as a control input for social networks is an idea worth exploring, in our opinion.

Our analysis is based on a synthetic network model and we run simulations in order to, at least, derive some qualitative general observations. Network behaviors observed in our tests are in line with theoretical studies on complex network controllability and some detailed investigations of our simulations have highlighted the specific mechanisms modifying network dynamics. Furthermore, considering that in practical situations it is often impossible to either recognise all theoretical driver nodes or accessing them with external perturbations, we have presented some empirical solutions based on network centrality metrics for selecting nodes that might have practical usage.

The problem of controlling social networks presents striking differences with respect to the study of structural controllability for complex industrial networks. The social context, in particular, introduces many limitations (ethical, operational, functional), but often does not strictly require full structural controllability. For these reasons, the application of control theory to social networks requires important adaptations. However, the results of our work look promising to us and encourage more analyses, tests, and verifications with respect to real social networks.

There are many situations in which it would be important to know how to handle the level of random information that agents receive. For instance, in learning situations, in social media, journalism, knowledge diffusion, skill acquisition, experience dissemination, immunisation from threats, and possibly risk management. In all these situations, there could be the problem of an excessive homophily and polarization (of interests, attention, analyses), but the solution cannot be to simply change what individuals prefer or believe or regard as important/interesting. Increasing information heterogeneity and serendipity could be effective approaches for improving the controllability of social contexts.

References

- [1] Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering* **24**(5), 896–911 (2012)
- [2] Allodi, L., Chiodi, L., Cremonini, M.: Self-organizing techniques for knowledge diffusion in dynamic social networks. In: *Proceedings of Complex Networks Conference 2014 (ComNet14)*. Bologna, Italy (2014)
- [3] Bradley, K., Smyth, B.: Improving recommendation diversity. In: *Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science*, Maynooth, Ireland, pp.

- 85–94. Citeseer (2001)
- [4] Centola, D., Gonzalez-Avella, J.C., Eguiluz, V.M., San Miguel, M.: Homophily, cultural drift, and the co-evolution of cultural groups. *Journal of Conflict Resolution* **51**(6), 905–929 (2007)
 - [5] Cornelius, S.P., Kath, W.L., Motter, A.E.: Realistic control of network dynamics. *Nature communications* **4** (2013)
 - [6] Cowan, N.J., Chastain, E.J., Vilhena, D.A., Freudenberg, J.S., Bergstrom, C.T.: Nodal dynamics, not degree distributions, determine the structural controllability of complex networks. *PloS one* **7**(6), e38,398 (2012)
 - [7] Crandall, D., Cosley, D., Huttenlocher, D., Kleinberg, J., Suri, S.: Feedback effects between similarity and social influence in online communities. In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 160–168. ACM (2008)
 - [8] Cremonini, M.: Introducing serendipity in a social network model of knowledge diffusion. *Chaos, Solitons & Fractals* **90**, 64–71 (2016)
 - [9] Easley, D., Kleinberg, J., et al.: Networks, crowds, and markets: Reasoning about a highly connected world. *Significance* **9**, 43–44 (2012)
 - [10] Gao, J., Liu, Y.Y., D’Souza, R.M., Barabási, A.L.: Target control of complex networks. *Nature communications* **5** (2014)
 - [11] Goldstone, R.L., Gureckis, T.M.: Collective behavior. *Topics in Cognitive Science* **1**(3), 412–438 (2009)
 - [12] Golub, B., Jackson, M.O.: How homophily affects the speed of learning and best response dynamics (2012)
 - [13] Kleinsman, J., Buckley, S.: Facebook study: a little bit unethical but worth it? *Journal of Bioethical inquiry* **12**(2), 179–182 (2015)
 - [14] Lin, C.T.: Structural controllability. *IEEE Transactions on Automatic Control* **19**(3), 201–208 (1974)
 - [15] Liu, Y.Y., Barabási, A.L.: Control principles of complex networks. *arXiv preprint arXiv:1508.05384* (2015)
 - [16] Liu, Y.Y., Slotine, J.J., Barabási, A.L.: Controllability of complex networks. *Nature* **473**(7346), 167–173 (2011)
 - [17] McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: Homophily in social networks. *Annual review of sociology* pp. 415–444 (2001)
 - [18] Menichetti, G., Dall’Asta, L., Bianconi, G.: Network controllability is determined by the density of low in-degree and out-degree nodes. *Physical review letters* **113**(7), 078,701 (2014)
 - [19] Motter, A.E.: Networkcontrology. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **25**(9), 097,621 (2015)
 - [20] Newman, M.: *Networks: an introduction*. Oxford university press (2010)
 - [21] Newman, M.W., Sedivy, J.Z., Neuwirth, C.M., Edwards, W.K., Hong, J.I., Izadi, S., Marcelo, K., Smith, T.F.: Designing for serendipity: supporting end-user configuration of ubiquitous computing environments. In: *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques*, pp. 147–156. ACM (2002)
 - [22] Pariser, E.: *The filter bubble: What the Internet is hiding from you*. Penguin UK (2011)
 - [23] Tucker, C.E.: Social networks, personalized advertising, and privacy controls. *Journal of Marketing Research* **51**(5), 546–562 (2014)

A Multiplex Approach to Urban Mobility

A. Baggag, S. Abba, T. Zanouda, J. Borge-Holthoefer and J. Srivastava

Abstract Multilayer networks have been the subject of intense research in the recent years in different applications. However, in urban mobility, the multi-layer nature of transportation systems has been generally ignored, even though most large cities are spanned by more than one transportation system. These different modes of transport have usually been studied separately. It is however important to understand the interplay between different transport modes. In this study, we consider the multi-modal transportation system, represented as a multiplex network, and we address the problem of urban mobility in the transportation system, in addition to its robustness and resilience under random and targeted failures. Multiplex networks are formed by a set of nodes connected by links having different relationships forming the different layers of the multiplex. We study, in particular, how random and targeted failures to the transportation multiplex network affect the way people travel in the city. More specifically, we are interested in assessing the portion of the city covered by a random walker under various scenarios. We consider the public transport of London as an application to illustrate the proposed capacity analysis method of multi-modal transportation, and we report on the robustness and the resilience of the system. This study is part of a project to develop a computational framework to better understand and predict mobility patterns in the city of Doha once its ambitious metro system is deployed in 2019. The computational framework will help the city to efficiently manage the flow of people and intelligently handle capacity through different transportation modes, in particular during mega events such as Soccer World cup FIFA 2022. The proposed method is based on the study in [9], but with an efficient computational approach resulting in tremendous savings in computational time. It is scalable and lends itself to efficient implementation on parallel computers.

Abdelkader Baggag (e-mail: abaggaga@qf.org.qa)✉ · Sofiane Abbar (e-mail: sabbar@qf.org.qa) · Tahar Zanouda (e-mail: tzanouda@qf.org.qa) · Jaideep Srivastava (e-mail: jsrivastava@qf.org.qa)

Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha (Qatar)

Javier Borge-Holthoefer (e-mail: borge.holthoefer@gmail.com)

CoSIN3 – Complex Systems @ IN3, Universitat Oberta de Catalunya, Barcelona (Catalunya)

1 A multiplex model of multi-modal transportation

Transportation networks in big cities are naturally multi-modal, and as such commuters use different modes to move around within the city. For example they use the combination of the bus system and the metro system to go from one location to another. These different modes of transportation have usually been studied separately by means of spatial networks to be able to understand aspects of urban planning systems and their evolution, see e.g. [5]. However, it has been shown that the different modes of transportation are not independent, and that their coupling can be critical and can affect the global behavior of the system. It is, therefore, important to study the properties of the full multimodal, multi-layer transportation network in order to understand the behavior of the city and to avoid possible negative side-effects of urban planning decisions. Hence, the study of the coupling between the different modes will provide a better understanding of the complex system, and the impact of the introduction of a new mode of transportation. It will also help planners prioritize which routes to target for adoption, in particular during mega events.

Many physical realities can be modeled as sets of interconnected entities; and multi-layer networks are used as a representation of these complex systems. We therefore observe many dynamical processes being studied on top of these networks, such as diffusion processes [11, 21], synchronization [4, 13], percolation [2, 18], etc. We use, in particular, multiplex networks to provide the convenient conceptual framework, see e.g. [6, 7, 8, 9, 10, 12, 15, 16, 17, 19, 20], and random walks to study the mobility of commuters within a multimodal transportation network in a city. This will allow the development of optimal navigation strategies.

1.1 Multi-layer networks

Given a set of L layers, each representing a type of relationship and containing N nodes. The relationship is represented by an edge and can be anything depending on the complex system, e.g., in Social Computing, it can be “friendship” on one layer such as Skype and “professional” on another layer, such as LinkedIn. The nodes represent the components of the complex system, e.g., bus stations in the first layer, and metro stations in the second layer, etc., for the multimodal transportation system. Even though the layers are different from each other, but the commuters use both of them to move in a large city, and therefore it is important to represent their mobility by taking into account the coupling between layers. The multiplex network is therefore defined as a finite sequence of intra-layer graphs $\mathcal{G}^\alpha = (\mathcal{V}^\alpha, \mathcal{E}^\alpha)$ coupled with the inter-layer supra-graph $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$ where $\mathcal{V}_c = \cup_\alpha \mathcal{V}^\alpha$ and

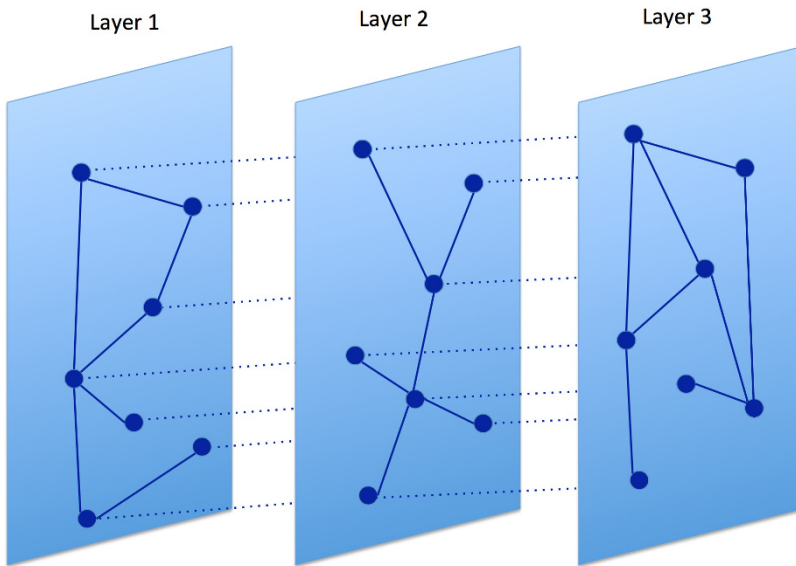
$$\mathcal{E}_c = \bigcup_{\alpha, \beta} \left\{ [i(\alpha), i(\beta)] \mid i(\alpha) \in \mathcal{V}^\alpha, i(\beta) \in \mathcal{V}^\beta, \alpha \neq \beta \right\},$$

where a node-layer $j(\alpha)$ means that node j participates in layer α . In this study, we consider node-aligned multiplex networks, i.e., inter-layer connections are “diagonal” in the sense that each node is connected only to its counterpart in the other layers,

and the inter-layer edges exist only between consecutive layers. Therefore the supra-adjacency matrix is block tri-diagonal and has the general form

$$\overline{\mathbf{W}} = \begin{bmatrix} \mathbf{W}^{(1)} + \mathbf{D}^{11} & \mathbf{D}^{12} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{D}^{21} & \mathbf{W}^{(2)} + \mathbf{D}^{22} & \mathbf{D}^{23} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{D}^{32} & \mathbf{W}^{(3)} + \mathbf{D}^{33} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \mathbf{D}^{(L-1)L} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}^{L(L-1)} & \mathbf{W}^{(L)} + \mathbf{D}^{LL} \end{bmatrix},$$

where $\mathbf{W}^{(\alpha)}$ is the adjacency matrix of layer α , $\mathbf{D}^{\alpha\beta}$ is a diagonal matrix such that $d_{ii}^{\alpha\beta}$ is the cost associated with the inter-layer edge $[i(\alpha), i(\beta)]$, and $\mathbf{D}^{\alpha\alpha}$ is a diagonal matrix such that $d_{ii}^{\alpha\alpha}$ represents the cost of staying in the same node and in the same layer.



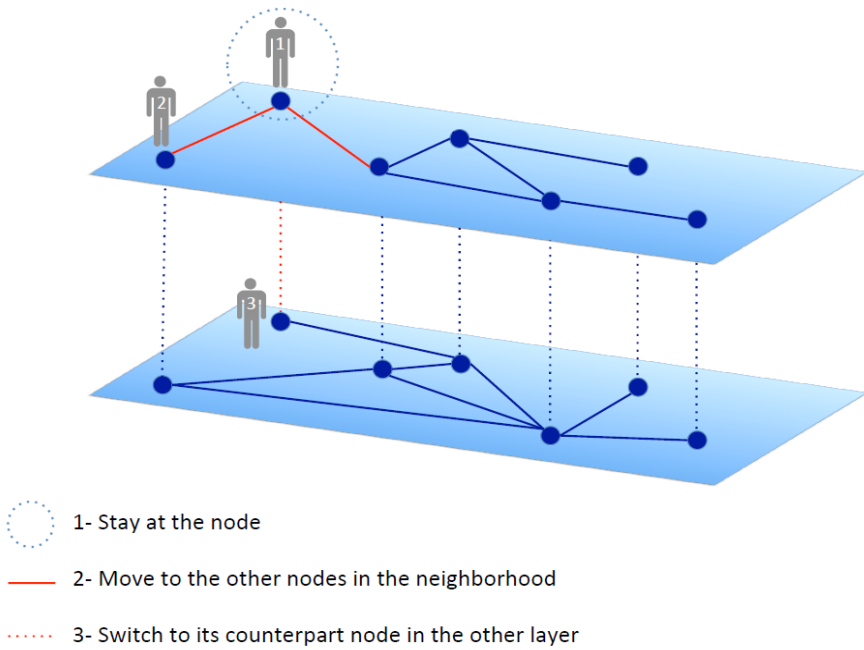
1.2 Random walk model of a citizen’s movement

Random walks constitute a fundamental mechanism for many dynamics taking place on complex networks. To assess the urban mobility in this multiplex transportation system, we represent the commuters as random walkers and we determine the coverage of the random walks, defined as the expected value of the number of steps to reach all nodes in the transportation system, regardless of the layer, on a walk that started from any node-layer $j(\alpha)$, i.e.,

$$\mathcal{C}_{j(\alpha)}(t) = \mathbb{E} \left[\# \text{ steps to reach all nodes in the graph on a walk that starts at } j(\alpha) \right],$$

i.e., it is the expected value of the number of nodes in the network being visited at least once in a time less than or equal to t , regardless of the layer, assuming that walks started from any other node-layer in the network.

A random walk is a Markovian process [22], which means that the transitions between states are historyless, i.e., the probability of transitioning to the next state depends only on the current state, not on any of the other previous states. Moreover, at each time step, the random walker has three options: the first one is to stay at the same node, the second one is to move to other neighboring nodes on the same layer and the last one is to switch to one of its counterparts on other layers, as illustrated in the figure below.



The mathematical model, developed in this paper, is inspired from the study in [9].

Therefore, given a multiplex transportation system of N nodes and L layers, the discrete-time master equation describing the probability of finding the walker in node-layer $i(\alpha)$, at time $(t + \Delta t)$, can be written as, e.g. see [9, 14]

$$\begin{aligned}
 p_{i(\alpha)}(t + \Delta t) &= \mathcal{A}_{ii}^{\alpha\alpha} p_{i(\alpha)}(t) + \sum_{j \neq i}^N \mathcal{A}_{ij}^{\alpha\alpha} p_{j(\alpha)}(t) + \sum_{\beta=1}^L \mathcal{A}_{ii}^{\alpha\beta} p_{i(\beta)}(t) \\
 &+ \sum_{\beta=1}^L \sum_{j \neq i}^N \mathcal{A}_{ij}^{\alpha\beta} p_{j(\beta)}(t)
 \end{aligned}
 \tag{1}$$

which can be assembled in matrix form as $\mathbf{P}(t + \Delta t) = \mathcal{A}\mathbf{P}(t)$, where $\mathcal{A} \in \mathbb{R}^{NL \times NL}$ is the transition supra-matrix (always assumed to be independent of time), and $\mathbf{P} \in \mathbb{R}^{NL}$

is a supra-vector containing the probability of finding the walker at any node-layer $i(\alpha)$, such that

$$\mathbf{P} = \left[\mathbf{p}_1^T \ \mathbf{p}_2^T \ \cdots \ \mathbf{p}_L^T \right]^T \quad \text{and} \quad \mathbf{P}\alpha = \left[p_{1(\alpha)} \ p_{2(\alpha)} \ \cdots \ p_{N(\alpha)} \right]^T.$$

For a classical random walk, the transition probability of moving from node-layer $i(\alpha)$ to node-layer $j(\alpha)$, i.e., within the same layer α , or to switch to the counterpart of vertex i in layer β , i.e., to node-layer $i(\beta)$, is uniformly distributed. Therefore we have

$$\mathcal{A}_{ij}^{\alpha\beta} = \begin{cases} \frac{d_{(i)}^{\alpha\alpha}}{k_{i(\alpha)} + c_{i(\alpha)}} & \text{if } i = j \text{ and } \beta = \alpha \\ \frac{w_{ij}^\alpha}{k_{i(\alpha)} + c_{i(\alpha)}} & \text{if } i \neq j \text{ and } \beta = \alpha \\ \frac{d_{(i)}^{\alpha\beta}}{k_{i(\alpha)} + c_{i(\alpha)}} & \text{if } i = j \text{ and } \beta \neq \alpha \\ 0 & \text{if } i \neq j \text{ and } \beta \neq \alpha \end{cases} \quad (2)$$

where w_{ij}^α is the weight of the intra-layer edge $[i(\alpha), j(\alpha)]$ and $d_{(i)}^{\alpha\beta}$ is the weight of the inter-layer edge $[i(\alpha), i(\beta)]$, i.e., the cost to switch from layer α to layer β at node i , while $d_{(i)}^{\alpha\alpha}$ quantifies the cost of staying in the same node and in the same layer. These are the elements of the matrices $\mathbf{W}^{(\alpha)}$, $\mathbf{D}^{\alpha\beta}$, and $\mathbf{D}^{\alpha\alpha}$ in $\overline{\mathbf{W}}$ respectively.

The intra-layer strength of a node-layer $i(\alpha)$ is $k_{i(\alpha)}$, and $c_{i(\alpha)}$ is the inter-layer strength of node i with respect to its connections to its counterparts in different layers. They are defined as

$$k_{i(\alpha)} = \sum_{j \in \mathcal{N}(i)} w_{ij}^\alpha \quad \text{and} \quad c_{i(\alpha)} = \sum_{\beta} d_{(i)}^{\alpha\beta},$$

so that the total strength of node-layer $i(\alpha)$ is the sum, i.e., $\kappa_{i(\alpha)} = k_{i(\alpha)} + c_{i(\alpha)}$.

Remark 1.1. Since each node is coupled only with its counterparts in different layers, then, only the elements of the type $\mathcal{A}_{ii}^{\alpha\beta}$ are different from zero. Jumps to other nodes in the other layers, as in Lévy random walks, are not allowed, and therefore $\mathcal{A}_{ij}^{\alpha\beta} = 0$.

2 Mathematical analysis of the model

In matrix form, and assuming that $\Delta t = 1$, it can be shown that the discrete-time master equation (1) can be written as the initial value problem,

$$\begin{cases} \frac{d}{dt} [\mathbf{P}(t)] = -(\mathcal{I} - \mathcal{A})\mathbf{P}(t), \\ \mathbf{P}(t=0) = \mathbf{P}(0) \end{cases} \tag{3}$$

and without loss of generality, we assume that, at $t = 0$, the random walker is in the first layer at node-layer $j(1)$, i.e., $\mathbf{P}(t = 0) = \mathbf{P}_j(0)$ then the initial value problem admits the following solution

$$\mathbf{P}(t) = \exp[-t(\mathcal{I} - \mathcal{A})]\mathbf{P}_j(0), \tag{4}$$

where $\exp[-t(\mathcal{I} - \mathcal{A})]$ is the usual matrix exponential, i.e.,

$$\exp[-t(\mathcal{I} - \mathcal{A})] = \sum_{k=0}^{\infty} \frac{(-t)^k}{k!} (\mathcal{I} - \mathcal{A})^k.$$

Remark 2.1. It is easy to see that $\mathbf{P}_j(0) = [\mathbf{e}_j^T \ \mathbf{0}^T \ \dots \ \mathbf{0}^T]^T$ with $\mathbf{e}_j \in \mathbb{R}^N$ being the canonical vector, and $\mathbf{0} \in \mathbb{R}^N$ is the vector of all zeros.

Theorem 2.1. *Let \mathcal{K} be the diagonal matrix containing the total strength of all nodes, i.e., $\mathcal{K} = \text{diag}(\overline{\mathbf{W}}\mathbf{1})$, where $\mathbf{1} \in \mathbb{R}^{NL}$ is the vector of all ones, then $\mathcal{A} = \mathcal{K}^{-\frac{1}{2}}\overline{\mathbf{W}}\mathcal{K}^{-\frac{1}{2}}$. Therefore, the matrix $(\mathcal{I} - \mathcal{A})$ is the normalized supra-Laplacian of the multiplex network.*

Proof. The supra-Laplacian of the multiplex network is

$$\begin{aligned} \mathcal{L} &= \mathcal{K} - \overline{\mathbf{W}} \\ &= \mathcal{K}^{\frac{1}{2}}(\mathcal{I} - \mathcal{K}^{-\frac{1}{2}}\overline{\mathbf{W}}\mathcal{K}^{-\frac{1}{2}})\mathcal{K}^{\frac{1}{2}} \end{aligned} \quad \square$$

The random walker can be at any layer, so let $p_i(t)$ be the probability to find the walker in node i at time t , regardless of the layer, i.e.,

$$p_i(t) = \sum_{\alpha=1}^L p_{i(\alpha)} = \mathbf{E}_i^T \mathbf{P}(t), \tag{5}$$

where $\mathbf{E}_i = [\mathbf{e}_i^T \ \dots \ \mathbf{e}_i^T]^T \in \mathbb{R}^{NL}$. Since $\mathbf{P}(t+1) = \mathcal{A}\mathbf{P}(t)$, and using Equations (5) and (4), we get at time $(t+1)$ the following expression for $p_i(t+1)$

$$\begin{aligned} p_i(t+1) &= \mathbf{E}_i^T \mathcal{A} \mathbf{P}(t) \\ &= \mathbf{E}_i^T \mathcal{A} \exp[-t(\mathcal{I} - \mathcal{A})]\mathbf{P}_j(0). \end{aligned} \tag{6}$$

To determine the coverage, defined as in [9], let's find an expression for the probability $\delta_{i,j}(t)$ not to find the walker in vertex i after t time steps, assuming it started in vertex j , that is

$$\delta_{i,j}(t) = [1 - p_j(0)] \prod_{\tau=1}^t [1 - p_i(\tau)]. \tag{7}$$

From (7), we get the recurrence relation $\delta_{i,j}(t + 1) = \delta_{i,j}(t) [1 - p_i(t + 1)]$, thus leading to the initial value problem

$$\begin{cases} \frac{d}{dt} [\delta_{i,j}(t)] = -\delta_{i,j}(t) \mathbf{E}_i^T \mathcal{A} \exp[-t(\mathcal{I} - \mathcal{A})] \mathbf{P}_j(0), \\ \delta_{i,j}(t = 0) = \delta_{i,j}(0), \end{cases} \tag{8}$$

with $\delta_{i,j}(0) = 0$ for $j = i$ since the walker started in vertex j and the probability of not finding it in the same vertex is 0. In the case of $j \neq i$, then $\delta_{i,j}(0) = 1$.

The solution to the initial value problem (8) is, see [9]

$$\delta_{i,j}(t) = \delta_{i,j}(0) \exp\left[-\mathbf{E}_i^T \mathcal{B} \mathbf{P}_j(0)\right] \quad \text{with} \quad \mathcal{B} = \sum_{\tau=0}^t \mathcal{A}^{\tau+1}. \tag{9}$$

Therefore, the coverage is given by double averaging over all vertices the probability $[1 - \delta_{i,j}(t)]$, i.e.,

$$\mathcal{C}(t) = 1 - \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j}(0) \exp\left[-\mathbf{E}_i^T \mathcal{B} \mathbf{P}_j(0)\right]. \tag{10}$$

Theorem 2.2. *The matrix \mathcal{B} need not be formed explicitly, since only its action on the vector $\mathbf{P}_j(0)$ is needed, i.e., a matrix-vector product, therefore*

$$\begin{aligned} \mathcal{B} \mathbf{P}_j(0) &= \left[\sum_{\tau=0}^t \mathcal{A}^{\tau+1} \right] \mathbf{P}_j(0) = [\mathcal{A} + \mathcal{A}^2 + \dots + \mathcal{A}^{t+1}] \mathbf{P}_j(0) \\ &= \mathcal{A} \mathbf{P}_j(0) + \mathcal{A} (\mathcal{A} \mathbf{P}_j(0)) + \dots + \mathcal{A} (\mathcal{A} \dots (\mathcal{A} \mathbf{P}_j(0)) \dots) \end{aligned}$$

Moreover, since $\mathbf{P}_j(0) = [\mathbf{e}_j^T \mathbf{0}^T \dots \mathbf{0}^T]^T$, then $\mathcal{A} \mathbf{P}_j(0) = [(\mathcal{A}(1 : N, j))^T \mathbf{0}^T \dots \mathbf{0}^T]^T$, i.e., the j th column of \mathcal{A} and we get the following recurrences

$$\begin{aligned}
\mathbf{E}_i^T \mathcal{A} \mathbf{P}_j(0) &= \mathcal{A}(i, j) \\
\mathbf{E}_i^T \mathcal{A}^2 \mathbf{P}_j(0) &= \sum_{\ell_1=1}^N \mathcal{A}(i, \ell_1) \mathcal{A}(\ell_1, j) \\
\mathbf{E}_i^T \mathcal{A}^3 \mathbf{P}_j(0) &= \sum_{\ell_1=1}^N \sum_{\ell_2=1}^N \mathcal{A}(i, \ell_1) \mathcal{A}(\ell_1, \ell_2) \mathcal{A}(\ell_2, j) \\
\mathbf{E}_i^T \mathcal{A}^4 \mathbf{P}_j(0) &= \sum_{\ell_1=1}^N \sum_{\ell_2=1}^N \sum_{\ell_3=1}^N \mathcal{A}(i, \ell_1) \mathcal{A}(\ell_1, \ell_2) \mathcal{A}(\ell_2, \ell_3) \mathcal{A}(\ell_3, j) \\
&\vdots \\
\mathbf{E}_i^T \mathcal{A}^{t+1} \mathbf{P}_j(0) &= \sum_{\ell_1} \sum_{\ell_2} \cdots \sum_{\ell_t} \mathcal{A}(i, \ell_1) \mathcal{A}(\ell_1, \ell_2) \mathcal{A}(\ell_2, \ell_3) \cdots \mathcal{A}(\ell_t, j)
\end{aligned}$$

Proof. These relations can be proven easily the usual way of proving recurrences, i.e., validate for the initial case, then assume it is correct for τ and prove that it is still correct for $\tau + 1$. The details are skipped. \square

3 Computational approach

3.1 Existing approach

In [9], the general form of the coverage, based on the eigendecomposition of the normalized supra-Laplacian $(\mathcal{I} - \mathcal{A}) \in \mathbb{R}^{NL \times NL}$ has the following expression

$$\mathcal{C}(t) = 1 - \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \delta_{i,j}(0) \exp \left[- \sum_{\ell \in \mathbf{\Lambda}^0} C_{i,j}(\ell) t - \sum_{\ell \in \mathbf{\Lambda}^+} C_{i,j}(\ell) \frac{e^{-\lambda_\ell t} - 1}{-\lambda_\ell} \right], \quad (11)$$

where $C_{i,j}(\ell) = \mathbf{E}_i^T \mathcal{A} \mathbf{V}_\ell \mathbf{P}_j(0)$ are constants depending on the vertex, the transition matrix, the eigendecomposition, and the initial conditions. Each supramatrix \mathbf{V}_ℓ is obtained from products of the eigenvectors of the normalized supra-Laplacian, and $\mathbf{\Lambda}^0$ and $\mathbf{\Lambda}^+$ indicate the sets of all null and positive eigenvalues of the normalized supra-Laplacian, respectively.

Remark 3.1. Any solution approach based on the eigendecomposition is time consuming and hard to obtain, especially for large matrices. Therefore it should be avoided.

3.2 Proposed algorithm

The main kernel in computing the coverage in Equation (10) is how to compute the exponent $\mathbf{E}_i^T [\mathcal{A} + \mathcal{A}^2 + \cdots + \mathcal{A}^{t+1}] \mathbf{P}_j(0)$. For this, we propose Algorithm 16. Therefore, the way the coverage is computed here results in a tremendous saving in

the computational time, as opposed to the eigendecomposition of the (normalized) supra-Laplacian matrix $(\mathcal{I} - \mathcal{A})$ proposed in [9], see Equation (11).

Algorithm 16 Computing $\mathbf{E}_i^T \left[\mathcal{A} + \mathcal{A}^2 + \dots + \mathcal{A}^{t+1} \right] \mathbf{P}_j(0)$

```

1: procedure COMPUTEEXPONENT( $\mathcal{A}, N, L, i, j, t$ )
2:    $\mathbf{P}_j(0) \leftarrow \left[ \mathbf{e}_j^T \ \mathbf{0}^T \ \dots \ \mathbf{0}^T \right]^T$ 
3:    $\mathbf{a} \leftarrow \mathcal{A} \mathbf{P}_j(0)$  ▷  $j$ th column of  $\mathcal{A}$ 
4:    $\bar{\mathbf{a}} \leftarrow \mathbf{a}$ 
5:   for  $\tau \leftarrow 1, t$  do
6:      $\mathbf{a} \leftarrow \mathcal{A} \mathbf{a}$  ▷ 1 matrix-vector product per iteration
7:      $\bar{\mathbf{a}} \leftarrow \bar{\mathbf{a}} + \mathbf{a}$  ▷ vector update
8:   end for
9:   exponent  $\leftarrow 0$ 
10:  for  $\alpha \leftarrow 1, L$  do
11:    exponent  $\leftarrow$  exponent +  $\bar{\mathbf{a}}(i + (\alpha - 1)N)$ 
12:  end for ▷ exponent =  $\mathbf{E}_i^T \bar{\mathbf{a}}$ 
13: end procedure

```

4 Experimental evaluation

The main objective of this work is to study urban mobility challenges in modern cities, as well as the robustness and resilience of the complex transportation systems. The multilayer nature of the proposed framework requires data from different modes of transportation. This data is not, in general, readily available. Thus, we first validate our proposed method using random graphs. Then, we perform an experimental study on the big city of London.

In this section, we start by describing the process of collecting the data. Then, we explain how different data sources are merged in order to build a multiplex network. Finally, we demonstrate and discuss our results on random graphs and real data from London's transportation network.

4.1 Coverage on random graphs

We benchmark our framework by creating a multiplex using the following configuration of random graphs:

1. a two-layer Barabási-Albert graph with 100 nodes each, and 196 edges each but not the same set of edges; and

2. a two-layer multiplex with a Barabási-Albert random graph in the first layer (100 nodes and 196 edges) and an Erdős-Rényi graph in the second layer (100 nodes and 1497 edges).

Figure 1 plots the coverage results of our numerical method.

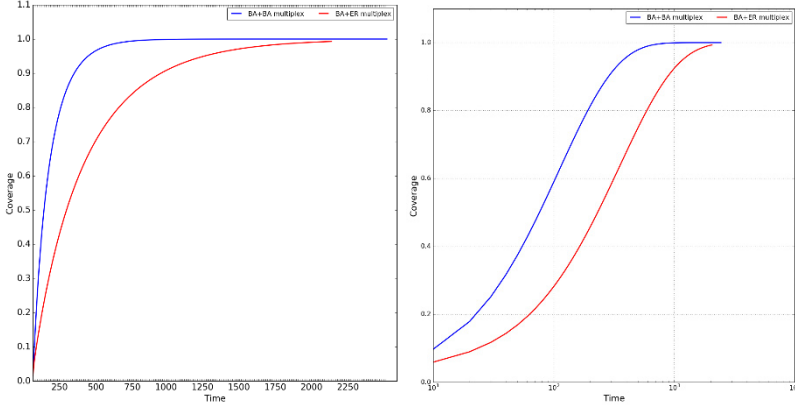


Fig. 1: Coverage over time for a two-layer multiplex Barabási-Albert + Barabási-Albert, and Barabási-Albert + Erdős-Rényi, with $d_{(i)}^{\alpha\alpha} = d_{(i)}^{\alpha\beta} = 1$

4.2 Coverage and resilience on real data

Accessing real transportation data is crucial to this study. However, due to the unavailability of such data, we limited our experiment to the city of London that has several open data portals available to the research community. Our data comes from two sources: OpenStreetMap (OSM)¹ and the National Public Transport Data (NPTDR) [1]. OSM provides an updated map of different bus and metro stations in the city, whereas NPTDR contains a snapshot of every public transport journey in Great Britain for a selected week in October each year. We represent the transportation network as a two-layer multiplex: Bus network and Metro network. The nodes of each layer represent the stations (bus stations in layer one, metro stations in layer two.) It is worth noticing that these two transportation modes are the most significant in the urban system of London. The edges are the routes connecting nodes (bus/metro stations). In order to establish a connection (a link) between a node in one layer (e.g., bus) and its counterpart in the other layer (e.g., metro), we adopted the simple assumption according to which two nodes in two different layers that are within a walking distance radius (≤ 100 m) are the same.

While NPTDR database covers Great Britain (England, Scotland, Wales), we focus only on London city. First, we filter all the stations from NPTDR that are

¹ <http://www.openstreetmap.org>

inside the bounding box of London city. Second, we extract all the stop points and trajectories of the two modes of transportation considered. Next, we use these stop points and trajectories to build the graph of each layer. Finally, we identify the inter-layer edges that connect all the same nodes residing in both layers.

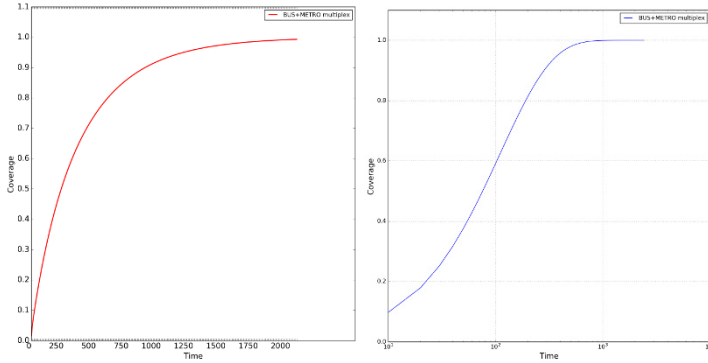


Fig. 2: Coverage versus time for a two-layer multiplex of London city Bus+Metro

We plot in Fig. 2 the coverage over time observed in the city of London. The right-hand panel shows a log on the x axis to ease the detection of the phase transition.

To quantify the robustness of the multimodal transportation system, we use percolation theory [3] to describe the impact of edge failures in the multiplex on the coverage. We iteratively remove edges from the multiplex and compute the new coverage of the resulting network. Figure 3 shows the degradation of the coverage function of the amount of removed edges. The panel to the left reports results of a random multiplex whereas the panel to the right reports results of the London network. Because of the stochastic nature of failures, we plot the average scores of 10 different runs. The key observation is that London transportation multiplex is quite robust as the removal of 70% of its edges leads to less than 20% loss of coverage.

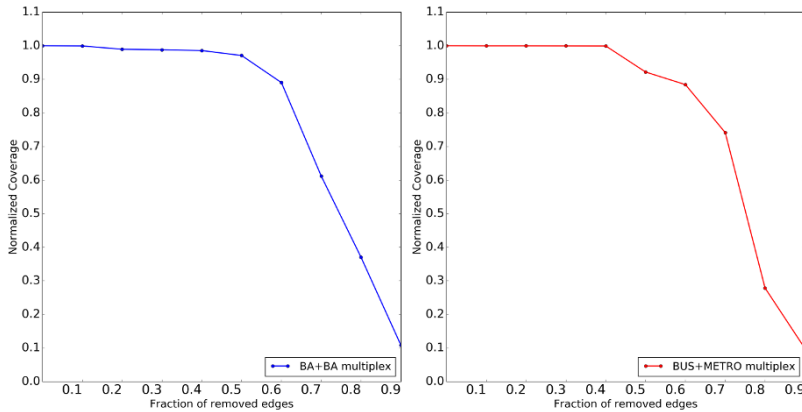


Fig. 3: Resilience of the (small) multiplex Barabási-Albert + Barabási-Albert, and of the London multiplex Bus+Metro

5 Conclusion

One of the critical areas is transportation, and the specific focus of the present study is on multi-modal transportation. The specific goal of this paper is (i) to build a mathematical model of the multi-modal transportation network as a mathematical structure called multiplex networks, and (ii) to simulate the commuters mobility in multiplex networks as a random walker, to study it as a Markovian process.

To better understand and predict mobility patterns in the city, we are working on a scalable computational framework that will help the city to efficiently manage the flow of people and intelligently handle capacity through different transportation modes. The proposed model has been validated and can be used to understand the underlying structure of urban mobility infrastructure of any city, using public data. This tool will help Doha to identify early problems, predict failures and design better transportation infrastructure.

References

- [1] National public transport data repository (2011). data.gov.uk/dataset/nptdr
- [2] Achlioptas, D., DSouza, R.M., Spencer, J.: Explosive percolation in random networks. *Science* **323**(5920), 1453–1455 (2009)
- [3] Albert, R., Jeong, H., Barabási, A.L.: Error and attack tolerance of complex networks. *nature* **406**(6794), 378–382 (2000)
- [4] Arenas, A., Diaz-Guilera, A., Kurths, J., Moreno, Y., Zhou, C.: Synchronization in complex networks. *Physical Reports* **469**(2), 93–153 (2008)
- [5] Barthélemy, M.: Spatial networks. *Physics Reports* **499**(1), 1–101 (2011)

- [6] Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C.I., Gómez-Gardeñes, J., Romance, M., Sendiña-Nadal, I., Wang, Z., Zanin, M.: The structure and dynamics of multilayer networks. *Physics Reports* **544**(1), 1–122 (2014)
- [7] De Domenico, M., Nicosia, V., Arenas, A., Latora, V.: Structural reducibility of multilayer networks. *Nature communications* **6** (2015)
- [8] De Domenico, M., Solé-Ribalta, A., Cozzo, E., Kivelä, M., Moreno, Y., Porter, M.A., Gómez, S., Arenas, A.: Mathematical formulation of multilayer networks. *Physical Review X* **3**(4), 041,022 (2013)
- [9] De Domenico, M., Solé-Ribalta, A., Gómez, S., Arenas, A.: Navigability of interconnected networks under random failures. *Proceedings of the National Academy of Sciences* **111**(23), 8351–8356 (2014)
- [10] De Domenico, M., Solé-Ribalta, A., Omodei, E., Gómez, S., Arenas, A.: Ranking in interconnected multilayer networks reveals versatile nodes. *Nature communications* **6** (2015)
- [11] Gómez, S., Diaz-Guilera, A., Gómez-Gardēnes, J., Perez-Vicente, C.J., Moreno, Y., Arenas, A.: Diffusion dynamics on multiplex networks. *Physical review letters* **110**(3) (2013)
- [12] Gómez, S., Diaz-Guilera, A., Gómez-Gardēnes, J., Perez-Vicente, C.J., Moreno, Y., Arenas, A.: Diffusion dynamics on multiplex networks. *Physical review letters* **110**(2), 028,701 (2013)
- [13] Gómez-Gardēnes, J., Gómez, S., Arenas, A., Moreno, Y.: Explosive synchronization transitions in scale-free networks. *Physical review letters* **106**(12) (2011)
- [14] Guo, Q., Cozzo, E., Zheng, Z., Moreno, Y.: Levy random walks on multiplex networks. arXiv preprint arXiv:1605.07587 (2016)
- [15] Lee, K.M., Min, B., Goh, K.I.: Towards real-world complexity: an introduction to multiplex networks. *The European Physical Journal B* **88**(2), 1–20 (2015)
- [16] Menichetti, G., Remondini, D., Panzarasa, P., Mondragón, R.J., Bianconi, G.: Weighted multiplex networks. *PloS one* **9**(6), e97,857 (2014)
- [17] Min, B., Do Yi, S., Lee, K.M., Goh, K.I.: Network robustness of multiplex networks with interlayer degree correlations. *Physical Review E* **89**(4), 042,811 (2014)
- [18] Radicchi, F., Fortunato, S.: Explosive synchronization transitions in scale-free networks. *Physical review letters* **103**(16) (2009)
- [19] Shai, S., Dobson, S.: Coupled adaptive complex networks. *Physical Review E* **87**(4), 042,812 (2013)
- [20] Solé-Ribalta, A., De Domenico, M., Kouvaris, N.E., Diaz-Guilera, A., Gomez, S., Arenas, A.: Spectral properties of the laplacian of multiplex networks. *Physical Review E* **88**(3), 032,807 (2013)
- [21] Watts, D.J.: A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences* **99**(9), 5766–5771 (2002)
- [22] Wilson, R.J.: An introduction to graph theory

Part VII
Network Visualization

Efficient Genealogical Graph Layout

Radek Marik

Abstract While a visual unconstrained tree structure planar layout design is easy to implement, a visualization of a tree with constraints on node ranks and their ordering within ranks leads to a difficult combinatorial problem. A genealogical graph, such as family tree, can be taken as an example of such a case. Classical ancestor trees, descendant trees, Hourglass charts, and their visual variants such as node-link diagrams or fan charts are suitable for assessment of peoples relationships when one is focused on a particular person and his/her direct ancestors and descendants. Such tree-based representations miss a broader context of relationships and do not allow the quick assessment of several interlinked families together. We propose a new undirected tree-driven layout technique for layered multitree graph visualizations producing constraints on node layers and ordering of groups of nodes within layers. The computed constraints can be mapped, at least partially, into the DOT language property directives used by the Graphviz toolbox. We demonstrate achievements on several datasets containing up to 39000 people.

1 Introduction and Related Methods

Although it is more than 55 years since Tutte introduced barycentric embedding, research of graph visualization techniques remains a highly active field attracting a lot of attention [16, 34, 35]. Graph visualization can help to form an overview of relational patterns and detect data structure much faster than data in a tabular form. The form in which the graph is presented has a significant impact on how the graph is understood and the time that is necessary to achieve this. Nodes placed close to one another might be interpreted by the user as a true relationship whether or not this

Radek Marik (e-mail: Radek.Marik@fel.cvut.cz)✉

Department of Telecommunication Engineering, Faculty of Electrical Engineering, Czech Technical University in Prague, Technicka 2, Dejvice, Prague, CZ-166 27, Czech Republic

WWW home page: <https://comtel.fel.cvut.cz/en/users/marikr>

relationship exists [16, 26]. Working with genealogical graphs is no exception in this sense.

Tree based drawing methods of genealogical graphs have been among the standard techniques for centuries. Ancestor trees, descendant trees and Hourglass charts belong to a set of traditional tools [21] implemented by a majority of freeware, shareware, or commercial tools, for example Gramps [1] or MyHeritage [3]. These tools provide a clear description of a situation when the user needs to investigate direct ancestors and/or descendants of a given person (often referred to as the main or center person) placed into the root of the tree. Thus, the generation of the main person consists of only one person and the size of other generations grows exponentially with a branching factor often over 2. Therefore, the classical node-link tree graphical representation resulting in a triangular shape wastes about one half of the drawing area. There are other more space-efficient representations such as fan charts or H-charts [5, 22, 36, 38]. As any pure tree representation enables any ordering of node predecessors/successors, it is possible to specify the type of ordering, such as children ordered by their birth dates. It is also possible to extend any such tree representation with additional nodes that can be attached as single nodes to any tree node (in the Gramps tool [1] this type of graph is called a Relationship Graph). In this way a tree with direct ancestors/descendants can cover, for example, spouses/partners. Therefore, tree representations can be laid out in such a way that family members are grouped together. The obvious drawback of the pure tree representations is that selecting a different main person leads to a different graph that must be rendered again. Such tree-based representations also miss a broader context of relationships and do not allow the quick assessment of several interlinked families together.

However, the situation with family member grouping changes significantly if the assumptions of one main person and direct ancestors/descendants are dropped. In a number of cases it is highly beneficial if the entire network of families, or at least a significant part, can be displayed in one layout. Then we face issues with challenges linked to edge crossing and preferences on node clustering [32, 33, 37]. Therefore, the standard techniques for planar graph layouts [6, 19, 20, 23, 29, 31] including planarization techniques [8, 9, 25, 30] are not suitable in all cases. Methods designed for layered graphs aimed at exact solutions [37] do not scale to large graphs; methods related to two-layer crossing problems using either averaging heuristics, such as the barycenter and the median methods [17, 33], or a hybrid approach [13, 15], produce layouts far from the optimum; and a local node order propagation [24] cannot resolve more global node order constraints.

The genealogical tools often use methods proposed for a general graph layout, such as hierarchical layouts, for example, implemented and provided by tools such as `dot.exe` (DOT) in Graphviz package [2] or yEd [4]. Unfortunately, these tools, and others we are aware of, do not support any kind of constraints that would allow the setting of node cluster preferences.

Assuming that a genealogical graph is layered according to the generation levels determined by an algorithm, such as the one proposed in the next section, the main complaint stems from mixing of children/partners from different families. Based on our own experience and observations made during our cooperation with

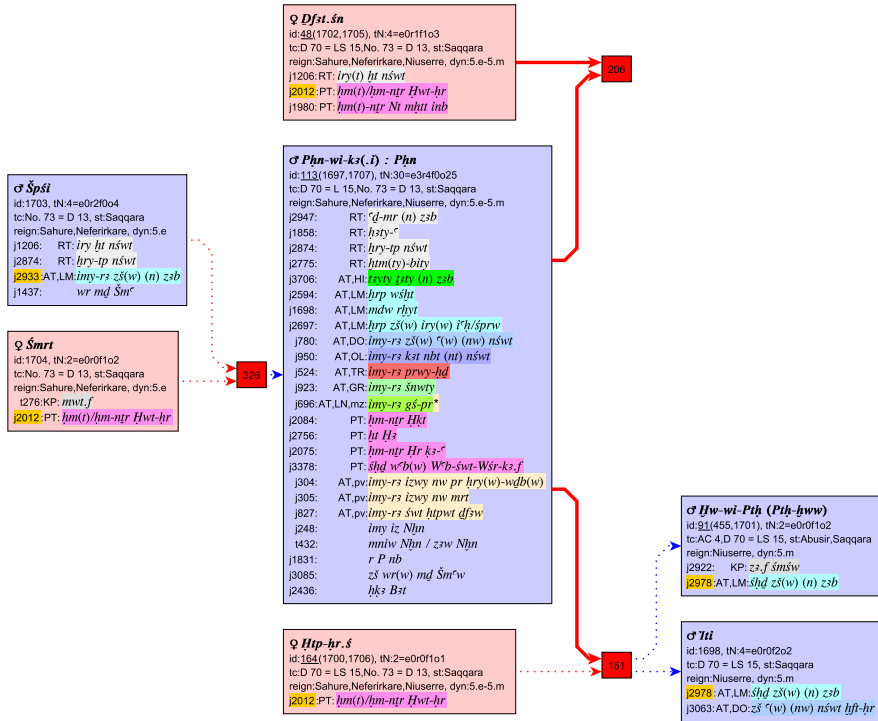


Fig. 1: A family tree component presented using a tree layout which is illustrative of Vizier Pehenuikas family. The people rectangles contain information such as their titles and offices.

Egyptologists, the researchers prefer grouping based on families. For example, Fig. 1 depicts Vizier Pehenuikas family reconstructed from the database of the Egyptian officials [12]. In this case, the layout was produced using the yEd tool.

When several families linked through a partnership relationship are visualized, one can cluster either children or partners, but generally not both. For example, Relationship graph visualization implemented in the DOT creates subgraphs of partners. Unfortunately, directed hierarchical drawing methods such as the very good one implemented as dot.exe [15] results in layouts mixing generations and members of several families, see Fig. 2. Children are often ordered in families randomly, furthermore children might be assigned to different ranks, children from different families might be interleaved, and a number of edge crossings occur. Such layouts are difficult to read and comprehend.

Graph specifications do not contain usually any constraints on node layers. Layer layout implementations rank nodes as proposed by many authors [15, 33]. In many situations the resulting layout is produced as required. Unfortunately, general criteria lead to node placement breaking generation layering as is usual and expected in genealogical graphs, i.e. children of one family at the same level and similarly their parents, see Fig 2. Some implementations, such as the DOT language, enable a

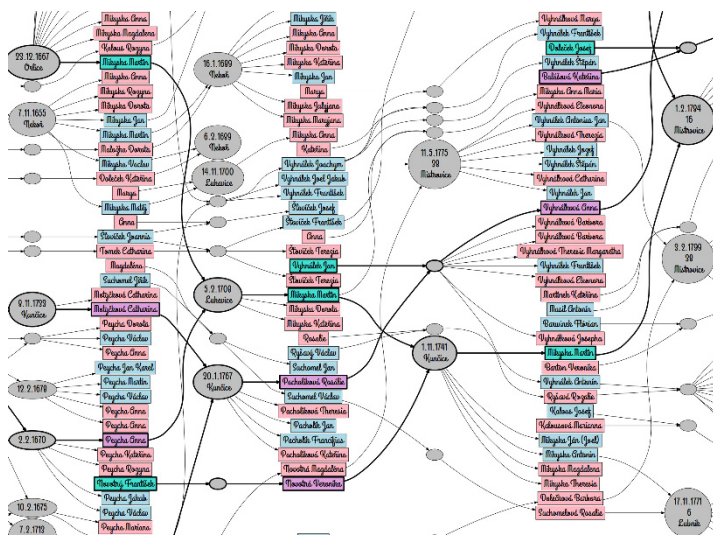


Fig. 2: A sample partial snapshot of a private family tree consisting of 2192 people as rendered using the DOT tool without any further constraints. Colored rectangles represent people (reddish for women, blueish for men). Ovals capture their marriages. Although the visualization seems to be correct, there are many cases when people are moved into different generation layers, many children from different families are mixed, and a number of edge crossings can be avoided.

specification that a subset of nodes shares the same layer (rank). The majority of algorithms computing ranks are derived from the topological order computation ($O(n)$ time complexity) [10] and select one of many possible solutions that satisfy layer intervals of node placements. Classical algorithms start from a single node, the only one with no predecessors. Generally, a genealogical graph can consist of several nodes without predecessors and several nodes without successors.

Formally, a genealogical graph is an acyclic bipartite directed graph $G(V_P, V_M, E)$ with two sorts of nodes, people V_P and marriages/partnerships V_M . The edges E are directed from parent nodes to marriage nodes and from marriage nodes to children nodes. If a family tree contains multiple marriages from one family to another, but it does not contain marriages between any two blood relatives, then it forms a multitree [14, 27]. A layering of an acyclic digraph $G(V_P, V_M, E)$ is a partition of $V_P \cup V_M$ into subsets L_1, L_2, \dots, L_h , such that if $(u, v) \in E$, where $u \in L_i$ and $v \in L_j$, then $i < j$ [7]. Without loss of generality we can assume that the index of the generation layer of parents (also denoted as ranks) is lower than the index of their marriage node, and further that the index of the marriage node is lower than the index of children nodes. In this paper, a (total) node order reflects a linear sequence of nodes in a given layer.

We are not aware of any method that would enable the definition and use of the necessary layout constraints. Recently, it was demonstrated using two simple propagated node order constraints that node layouts of such graphs can be improved

significantly [24]. The proposed topological layout technique in [24] is based on a local propagation of children and parent ordering across generations (ranks), but it is not able to reflect global subtree constraints.

In this paper we propose a new method that allows the determination of such node order constraints using an undirected tree-driven layout of subtrees and does not exhibit deficiencies of the local propagation [24]. The new method produces significantly fewer edge crossings than the methods mentioned above. At least partially, the proposed constraints can be mapped to additional graph specifications that result in the DOT algorithm producing the required layout. More specifically, our method modifies the first two steps of the approach proposed in [15, 33], i.e. 1/ determination of layers (generations, node ranks, levels), and 2/ enforcing node orders within the layers. In provided visualizations, layers define a horizontal index while orders are reflected by a vertical index.

The rest of the paper is organized in the following way. In the next section we present an algorithm that allows setting layers of nodes for an acyclic graph representing a traditional representation of family tree using marriage nodes. Then we describe the technical details of the new proposed method calculating node orders within the layers. Finally, we discuss achieved results and tests on real data datasets with thousands of nodes.

2 Layering of Genealogical Graph Nodes

In this section we present an algorithm already proposed in [24], using which the ranks of nodes can be determined for any genealogical graph. In the algorithm we assume that the processed graph is directed and acyclic. Let us use a convention that node ranks are identified by numbers $\lambda_2(v)$ and successors have higher levels. Each node is assigned an interval of rank levels at which the node can appear with regard to a base level. The algorithm uses two simple passes through a graph. Each node is assigned the highest possible level with respect to the current highest base level of successors during the first pass. In fact, this pass assigns node layers conforming to longest path layering [17].

$$\lambda_1(v) = \begin{cases} \max_{(v,w) \in E} \lambda_1(w) - 1 & \text{if } v \text{ has successors} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Thus, the node(s) with the lowest level can be determined. A rank level for each node is set as the maximum level of the node predecessor levels increased by one during the second pass. The second pass starts from the nodes with the lowest level.

$$\lambda_2(v) = \begin{cases} 0 & \text{if } v \text{ has the lowest level} \\ \lambda_2(w) - 1 & \text{if } w \text{ has predecessors partially processed} \\ & (v, w) \in E \text{ and } \lambda_2(v) \text{ is not assigned} \\ \min_{(w,v) \in E} \lambda_2(w) + 1 & \text{if } v \text{ has all predecessors processed} \end{cases} \quad (2)$$

Each node is visited twice during each pass using depth first search (DFS) using an explicit LIFO queue. The first visit ensures that all successors/predecessors are processed already. When the node is visited again, its level is determined as minimum/maximum of successors/predecessors levels. As children from a single marriage have only one common predecessor, the marriage node, they share the same rank level. However, parent nodes can be assigned to different levels. Nevertheless, the algorithm guarantees that parents linked to a marriage node always have a lower layer number than the marriage node and children attached to the marriage node have higher layer numbers than the marriage node. The algorithm uses two DFS passes with linear time complexity $O(N)$, where N is the number of graph nodes.

3 Nodes Ordering within Layers

In this section we support an approach that results in siblings of one family being clustered tightly while partnerships/parents might be mixed. The obvious reason behind this variant is that the number of children is much higher than 2, often reaching values over 10. Thus, an injected edge crossing because of mixed parents is much lower than occurs when children are mixed, and families can be identified easily by a number of parallel edges leading from marriage nodes to children nodes.

The problem of a layout design might then be reduced to a determination of the order of people belonging to one generation layer. We propose that children belonging to a single family are ordered by their birth dates. Subtrees of the child descendants, including descendant marriage nodes, hold this order. In the opposite direction, i.e. from a marriage node to its spouses, the order of spouses can be determined according to birthdates of spouses. There might be cases when two or more people from two or more different families create partnerships. In such situations we cannot insist on the order of marriage nodes as the order requirements might be contradictory, for example, in the case of two families both with two children that creates two marriages in the opposite order of their birthdates.

As cases when two individuals share two or more distinct subtrees are very rare in reality (just 5 cases in our database of 2192 individuals), we can transform the genealogical graph into an undirected tree by removing a few edges. The undirected tree can be decomposed into subtrees. These subtrees are layered recursively as strips side by side while following a simple set of rules that minimize edge crossings. Nodes of processed subtrees are placed into rank arrays during the subtree layering. Thus, the rank array determines the order of their nodes. The whole method can be described using four steps:

1. Node rank determination
2. Undirected spanning tree subgraph selection
3. Computation of shape characteristics for all subtrees
4. Node order design by subtrees layering.

The used node rank determination was already described in Sect. 2. Thus, we are going to focus on the other three steps. Structures driving node order design are created in steps 2 and 3. The actual layout is performed in the last step.

3.1 Undirected spanning tree subgraph selection

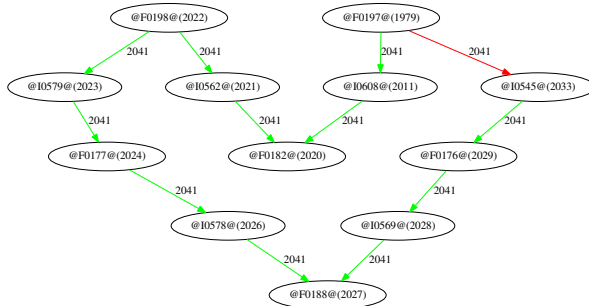


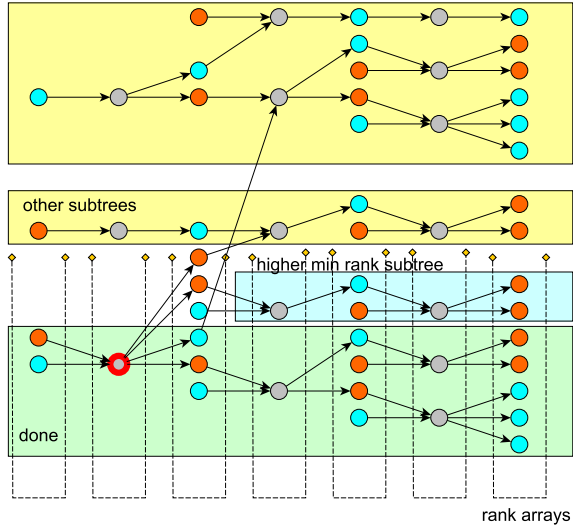
Fig. 3: A detected block of the sample family tree database. The backedge is red. One can deduce from the block that descendants of the marriage node @F0188@ share ancestors of two distinct subtrees @F0198@ and @F0197@.

The layout design is controlled by an undirected spanning tree of the original directed graph. Removing backedges using a DFS scan is a sufficient simple solution. These removed edges are drawn but not used by the node ordering algorithm. However, we might select other edges to stress their particular role in the graph because the removed edges typically cause edge crossings. We select suitable edges using blocks (biconnected components [11]) in linear time [18, 28] to break cycles (a DFS scan, cycles determined by backedges), see Fig. 3. Blocks might occur if two or more individuals share two or more distinct ancestor subtrees. A related analysis of the blocks is considered beyond the scope of this paper.

3.2 Subtree shape characteristics

As we will see later, the graph is layered starting from a node with the lowest rank. The layout technique makes decisions based on subtree orders (the number of nodes) and comparisons of the current node rank with the minimum node ranks of processed subtrees. Starting from the node with the lowest rank we assign both pre/post order timestamps τ_1, τ_2 to each node using the DFS scanning [10]. The order of any undirected subtree can be calculated as $(\tau_2^r - \tau_1^r)/2$, where τ_1^r, τ_2^r are the timestamps of the subtree root, because each node of the subtree has just two timestamps.

Fig. 4 A symbolical snapshot of the proposed layout method. The current node is tagged by its thick red border. Blue nodes represent men, orange nodes represent women, gray nodes represent their marriages. The greenish zone is an already processed part of the graph with all nodes registered in the rank arrays that keep their order of registration. The blue zone contains just one subtree with the minimum rank higher than the rank of the current node. Two yellow zones represent another two subtrees in the order in which they be layered based on their number of nodes.



Using the post order we can determine the minimum node rank $\Lambda(v_i)$ of any subtree determined by its root node v_i inside its timestamp interval $[\tau_1(v_i), \tau_2(v_i)]$ in $O(N)$. The $NB(v_i)$ function produces undirected neighbor nodes of v_i .

$$\Lambda(v_i) = \min(\lambda_2(v_i), \min_{v_j \in NB(v_i): \tau_2(v_j) < \tau_2(v_i)} \Lambda(v_j)) \tag{3}$$

3.3 Design of node order within layers

Again starting from the node with the lowest rank we assign nodes of timestamp interval subtrees into rank arrays (initially empty for each rank), see Fig. 4. First, subtrees with a minimum rank higher than the rank of the current node v_i are processed because their edges do not cross any other edges in the rest of the graph. Then the remaining subtrees are processed according to their increasing size to minimize edge crossing, because it is expected that an edge linking the current node to a subtree and crossing other larger graph subtrees produces more edge crossings. Let us assume we process the current node where some children nodes link K subtrees with a minimum rank lower than the current node rank. A sequence $[cr_1, \dots, cr_K]$ is obtained if the subtrees are sorted according to their edge crossing counts cr_ℓ between the children node and its subtree. If these subtrees are layered side by side and each child is linked with them then the total number of injected edge crossings is $CR_{v_i} = \sum_{j=2}^K \sum_{k=1}^{j-1} cr_k = \sum_{\ell=1}^{K-1} (K - \ell) cr_\ell$ that is minimum if the sequence $[cr_1, \dots, cr_K]$ is not decreasing.

The method can be described as the following sequence of steps:

1. Set an empty array for each node rank.
2. Set an empty LIFO stack of processed nodes.
3. Add the node with the lowest rank to the stack.

4. Pop a node v_i from the stack.
5. Register the node to the appropriate rank array given by $\lambda_2(v_i)$.
6. Register all children of v_i and their spouses with empty ancestor subtrees in the required order given by birthdates.
7. Select nodes $v_j \in NB(v_i)$ with $\Lambda(v_j) \leq \lambda_2(v_i)$ and add them to the stack sorted by the decreasing size of their subtrees given by $\tau_2(v_j) - \tau_1(v_j)$.
8. Select nodes $v_j \in NB(v_i)$ with $\Lambda(v_j) > \lambda_2(v_i)$ and add them to the stack sorted by birthdates.
9. If the stack is empty, then stop, otherwise continue with step 4.

The algorithm is a kind of DFS scanning with the linear complexity $O(N)$ at the top level. The selections of nodes in steps 7 and 8 can be performed in linear time, too. Steps 7 and 8 also performs sorting with complexity $O(N_v \log(N_v))$ where N_v is a branching factor which is a very low number in genealogical graphs. N_v is often limited by value 15 (a maximum number of children and two parents) so the complexity of steps 7 and 8 can be treated as a constant. Thus, the overall complexity of the layout method is close to $O(N)$.

4 Implementation, Experiments, and Discussion

We implemented the proposed first two steps of an acyclic genealogical graph layout algorithm. The steps produce the constraints on generation layers and node orders in each generation.

We selected 20 datasets for an evaluation of the proposed constraints contribution. The first dataset consists of 2192 people of the authors private family relationship genealogical graph. The set is created as a merge of several family trees ranging over 14 generations with the first records dated 1647. The second dataset consists of 3057 people of the database created by Egyptologists [12]. The database covers high ranking officials from the 4th, 5th, and 6th dynasties and their families. One can reconstruct over 160 families with up to 6 generations. The database has been filled over ten years. Generated graphs covering more families greatly help Egyptologists to assess quickly investigated social phenomena. Experiments with families of the Egyptian database did not exhibit any layout deficiencies as the families are quite simple and not larger than 50 family members. Similar results were obtained for the rest of the datasets that are GEDCOM files downloaded from the Internet with from 400 to 39,000 individuals.

Layout constraints generated using the method proposed in this paper are depicted in Fig 5. The layers of nodes were placed uniformly in the horizontal direction while their ordered nodes were placed uniformly in the vertical direction. The nodes were linked with straight-lined edges. Family clans are kept well separated. The layout is created very quickly (below 0.5 second with a Python script on DELL XPS 13 using an Intel i7 2GHz processor).

The method can be further improved. We have not considered the correct number of edge crossings, but only its estimate. The layout design adds subtree layers only on one side. It could utilize both sides of the current node. In fact, a better method could

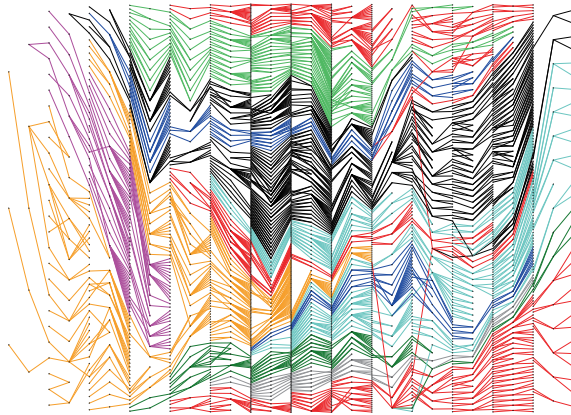


Fig. 5: A very low edge crossing visualization of the sample private family tree with 2192 individuals and 765 marriages created by the proposed method. Family clans with more than 150 people are emphasized with different colors. An ideal layout would result in edges creating *waves* only.

utilize a combinatorial assessment of subtree minimum ranks combined with subtree width at the current node rank. However, the complexity of such a method would be much higher while the gain in edge crossings would be minimal (considering the current datasets characteristics).

5 Conclusion

In this work we proposed a new method for a ranked multitree layout with constraints on node order and its layers. In fact, the constraints result in a fully specified topological arrangement of the graph nodes in plane. The constraints can be computed very efficiently. The experiments demonstrate clearly a significant improvement in graph comprehension because of low edge crossing and compact family grouping and indicate that the results provided by the present state of the art tools are quite far from the optimum layout, at least for special sorts of graphs such as genealogical ones. The smaller number of undirected backedges the better layout results.

Acknowledgements Sponsored by the project for GAČR, No. 16-072105: Complex network methods applied to ancient Egypt data in the Old Kingdom (27002180 BC).

References

- [1] Gramps. genealogical research software. <https://gramps-project.org/> (2016). Accessed: 5.6.2016
- [2] Graphviz - graph visualization software. www.graphviz.org (2016). Accessed: 5.6.2016
- [3] Myheritage. <https://www.myheritage.cz> (2016). Accessed: 5.6.2016

- [4] yed graph editor. <http://www.yworks.com/products/yed> (2016). Accessed: 5.6.2016
- [5] Ball, R., Cook, D.: A family-centric genealogy visualization paradigm. In: 14th Annual Family History Technology Workshop. Provo, Utah (2014)
- [6] Booth, K.S., Lueker, G.S.: Testing for the consecutive ones property, interval graphs and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences* **13**(3), 335–379 (1976)
- [7] Brandes, U., Köpf, B.: Fast and Simple Horizontal Coordinate Assignment, pp. 31–44. Springer Berlin Heidelberg, Berlin, Heidelberg (2002). DOI 10.1007/3-540-45848-4_3. URL http://dx.doi.org/10.1007/3-540-45848-4_3
- [8] Chimani, M., Gutwenger, C., Mutzel, P., Wong, H.M.: Upward planarization layout. *Journal of Graph Algorithms and Applications* **15**(1), 127–155 (2011)
- [9] Chimani, M., Junger, M., Schulz, M.: Crossing minimization meets simultaneous drawing. In: 2008 IEEE Pacific Visualization Symposium, pp. 33–40 (2008). DOI 10.1109/PACIFICVIS.2008.4475456
- [10] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, Third Edition, 3rd edn. The MIT Press (2009)
- [11] Diestel, R.: Graph Theory. Springer (2005)
- [12] Dulíková, V.: The reign of king Nyuserre and its impact on the development of the Egyptian state. A multiplier effect period during the Old Kingdom. Ph.D. thesis, Charles University in Prague, Faculty of Arts, Czech Institute of Egyptology (2016)
- [13] Eiglsperger, M., Siebenhaller, M., Kaufmann, M.: An Efficient Implementation of Sugiyama’s Algorithm for Layered Graph Drawing, pp. 155–166. Springer Berlin Heidelberg, Berlin, Heidelberg (2005). DOI 10.1007/978-3-540-31843-9_17. URL http://dx.doi.org/10.1007/978-3-540-31843-9_17
- [14] Furnas, G.W., Zacks, J.: Multitrees: Enriching and reusing hierarchical structure. In: Conference Companion on Human Factors in Computing Systems, CHI '94, pp. 223–. ACM, New York, NY, USA (1994). DOI 10.1145/259963.260396. URL <http://doi.acm.org/10.1145/259963.260396>
- [15] Gansner, E.R., Koutsofios, E., North, S.C., Phong Vo, K.: A technique for drawing directed graphs. *IEEE Transactions on Software Engineering* **19**(3), 214–230 (1993)
- [16] Gibson, H., Faith, J., Vickers, P.: A survey of two-dimensional graph layout techniques for information visualisation. *Information Visualization* **12**(3-4), 324–357 (2013). DOI 10.1177/1473871612455749. URL <http://ivi.sagepub.com/content/12/3-4/324.abstract>
- [17] Healy, P., Nikolov, N.S.: Handbook of Graph Drawing and Visualization, chap. Hierarchical Drawing Algorithms, pp. 409–454. CRC (2013)
- [18] Hopcroft, J., Tarjan, R.: Algorithm 447: Efficient algorithms for graph manipulation. *Commun. ACM* **16**(6), 372–378 (1973). DOI 10.1145/362248.362272. URL <http://doi.acm.org/10.1145/362248.362272>
- [19] Hopcroft, J., Tarjan, R.: Efficient planarity testing. *Journal of the ACM* **21**(4), 549–568 (1974). DOI 10.1145/321850.321852. URL <http://doi.acm.org/10.1145/321850.321852>
- [20] Hsu, W.L., McConnell, R.: Handbook of Data Structures and Applications, chap. PQ Trees, PC Trees, and Planar Graphs, pp. 32–1–32–27. CRC Press (2004)
- [21] Keller, K., Reddy, P., Sachdeva, S.: Family tree visualization. Course project report. http://vis.berkeley.edu/courses/cs294-10-sp10/wiki/images/f/f2/Family_Tree_Visualization_-_Final_Paper.pdf (2010). Accessed: 5.6.2016
- [22] Kieffer, S., Dwyer, T., Marriott, K., Wybrow, M.: Hola: Human-like orthogonal network layout. *IEEE Transactions on Visualization and Computer Graphics* **22**(1), 349–358 (2016). DOI 10.1109/TVCG.2015.2467451
- [23] Lempel, A., Even, S., Cederbaum, I.: An algorithm for planarity testing of graphs. In: P. Rosenstiehl, Gordon, Breach (eds.) *Theory of Graphs*, pp. 215–232. New York (1967)

- [24] Marik, R.: On large genealogical graph layouts (accepted for publication). In: WASACNA 2016 : Workshop on Algorithmic and Structural Aspects of Complex Networks and Applications, September 17th, Tatransk Matliare, Slovakia (2016)
- [25] Mathews, E., Frey, H.: A Localized Link Removal and Addition Based Planarization Algorithm, pp. 337–350. Springer Berlin Heidelberg, Berlin, Heidelberg (2012). DOI 10.1007/978-3-642-25959-3_25. URL http://dx.doi.org/10.1007/978-3-642-25959-3_25
- [26] McGrath, C., Blythe, J., Krackhardt, D.: Seeing groups in graph layouts. *Connections* **19**(2), 22–29 (1996)
- [27] McGuffin, M.J., Balakrishnan, R.: Interactive visualization of genealogical graphs. In: IEEE Symposium on Information Visualization, 2005. INFOVIS 2005., pp. 16–23 (2005). DOI 10.1109/INFVIS.2005.1532124
- [28] Paton, K.: An algorithm for the blocks and cutnodes of a graph. *Commun. ACM* **14**(7), 468–475 (1971). DOI 10.1145/362619.362628. URL <http://doi.acm.org/10.1145/362619.362628>
- [29] Reingold, E.M., Tilford, J.S.: Tidier drawings of trees. *IEEE Transactions on Software Engineering* **SE-7**(2), 223–228 (1981). DOI 10.1109/TSE.1981.234519
- [30] Resende, M.G.C., Ribeiro, C.C.: Graph planarization Graph Planarization, pp. 908–913. Springer US, Boston, MA (2001). DOI 10.1007/0-306-48332-7_187. URL http://dx.doi.org/10.1007/0-306-48332-7_187
- [31] Shih, W.K., Hsu, W.L.: A new planarity test. *Theoretical Computer Science* **223**(1-2), 179–191 (1999)
- [32] Sugiyama, K., Misue, K.: Visualization of structural information: automatic drawing of compound digraphs. *IEEE Transactions on Systems, Man, and Cybernetics* **21**(4), 876–892 (1991). DOI 10.1109/21.108304
- [33] Sugiyama, K., Tagawa, S., Toda, M.: Methods for visual understanding of hierarchical system structures. *IEEE Transactions on Systems, Man, and Cybernetics* **11**(2), 109–125 (1981). DOI 10.1109/TSMC.1981.4308636
- [34] Tutte, W.T.: Convex representations of graphs. *Proceedings of the London Mathematical Society, Third Series* (10), 304–320 (1960)
- [35] Tutte, W.T.: How to draw a graph. *Proceedings of the London Mathematical Society, Third Series* (13), 743–768 (1960)
- [36] Tuttle, C., Nonato, L.G., Silva, C.: Pedvis: A structured, space-efficient technique for pedigree visualization. *IEEE Transactions on Visualization and Computer Graphics* **16**(6), 1063–1072 (2010). DOI 10.1109/TVCG.2010.185
- [37] Warfield, J.N.: Crossing theory and hierarchy mapping. *IEEE Transactions on Systems, Man, and Cybernetics* **7**(7), 505–523 (1977). DOI 10.1109/TSMC.1977.4309760
- [38] Yoghourdjian, V., Dwyer, T., Gange, G., Kieffer, S., Klein, K., Marriott, K.: High-quality ultra-compact grid layout of grouped networks. *IEEE Transactions on Visualization and Computer Graphics* **22**(1), 339–348 (2016). DOI 10.1109/TVCG.2015.2467251

NodeTrix-Multiplex: Visual Analytics of Multiplex Small World Networks

Shivam Agarwal, Amit Tomar and Jaya Sreevalsan-Nair

Abstract Analyzing multiplex small world networks (SWNs) using community detection (CD) is a challenging task. We propose the use of visual analytics to probe and extract communities in such networks, where one of the layers defines the network topology and exhibits small-world property. Our novel visual analytics framework, NodeTrix-Multiplex (NTM), for visual exploration of multiplex SWNs, integrates focus+context network visualization, and analysis of community detection results, within the focus. We propose a heterogeneous data model, which composites multiple layers for the focus and context and thus, enables finding communities across layers. We perform a case-study on a co-authorship (collaboration) network, with a functional layer obtained from the author-topic similarity graph. We also perform an expert user evaluation of the tool, developed using NTM.

1 Introduction

Complex networks are real-world, ubiquitous and important, as networks can simultaneously encode objects in a specific context and the pairwise relationships between those objects. Small world networks (SWNs) are a class of complex networks [1, 31], which shows small-world property. Social networks, such as collaboration networks, are SWNs. Owing to the advances in technological capability of gathering, storing, and analysis of these data sets, such networks are increasingly encoding more information. Thus, the rich data is stored as multiplex complex networks, where different relationships, between the same set of nodes, are stored as separate *layers*. The layers of the multiplex network have unique adjacency matrices [3, 16]. Since our focus is on multiplex SWNs, we assume one of the layers in the network gives the network topology of a SWN, which in turn determines an initial community formation. We

Shivam Agarwal (e-mail: shivam.agarwal@iiitb.org) · Amit Tomar (e-mail: amit.tomar@iiitb.org) · Jaya Sreevalsan-Nair (e-mail: jnair@iiitb.ac.in)
Graphics-Visualization-Computing Lab, International Institute of Information Technology Bangalore, Karnataka, India.

call such a layer “*structural*” layer, and the other layers, such as similarity graphs, “*functional*” layers, borrowing terminology from brain networks [20]. Another way to look at it is that, we use the *existential* layer (i.e. the layer that has caused the very existence of the complex network) as the structural network, e.g. collaboration network. Thus, the other layers are “functional,” which depend on the existential layer. In the case of multiplex SWNs, we consider the existential layer, that exhibits the small-world property, to be the structural one.

Community detection (CD) can reveal several patterns in a complex network. However, CD across multiple layers is challenging owing to the differences in “*percolation*” of communities in the layers [8]. Here, we focus in selectively exploring the dynamics of communities within a small subnetwork in the complex network, which is a community in itself. Thus, for community exploration and detection in multiplex SWNs, we propose a **focus+context paradigm**, and a visual analytic framework, **NodeTrix-Multiplex** (NTM), that enables the user to see clustering tendencies in the focus. Visual analytics is an active area of research where visualization plays a larger role in data analytics, in an interweaved manner, than just summarizing information or exploring data. Figure 1 summarizes our proposed work, which shows **our proposed heterogeneous data model** (HDM), on which visual analytics is used for *drilling down* across layers in a subnetwork of interest. Our proposed visual analytic framework is designed with the visual information seeking mantra: *overview first, zoom and filter, then details on demand* [27]. NTM uses the hybrid representation of SWNs, as proposed in NodeTrix [13], which exploits the “locally dense, globally sparse” structure of a SWN. A preliminary version of our tool¹ is available at <http://nmultiplex.au-syd.mybluemix.net/>

Notations: We denote a multiplex network with N layers (each defined by a unique adjacency matrix), as $\mathcal{M} = \{\mathcal{V}(\mathcal{M}), \mathcal{E}^0, \dots, \mathcal{E}^{N-1}\}$, where $\mathcal{V}(\mathcal{M})$ is the vertex set of the network, and \mathcal{E}^i is the set of edges belonging to the i^{th} layer, and it is represented by the weighted adjacency matrix of the i^{th} layer. $e(u, v)$ implies an edge exists between vertices $u, v \in \mathcal{V}$ and it encodes the edge weight, a real value.

The i^{th} layer of \mathcal{M} is defined as $\mathcal{L}^i = \{\mathcal{V}(\mathcal{M}), \mathcal{E}^i\}$. Non-overlapping (or crisp) communities in any layer \mathcal{L}^i , are denoted as $\{\mathcal{C}_0^i, \dots, \mathcal{C}_{M_i-1}^i\}$ for M_i communities, where \mathcal{C}_j^i is the vertex set of the j^{th} community in the i^{th} layer. Thus, $0 \leq i < N$ and $0 \leq j, k < M_i$ where $j \neq k$, we get $\mathcal{V}(\mathcal{C}_j^i) \subset \mathcal{V}(\mathcal{M})$ and $\mathcal{V}(\mathcal{C}_j^i) \cap \mathcal{V}(\mathcal{C}_k^i) = \emptyset$.

Any subnetwork in \mathcal{L}^k is given as $\mathcal{N}(k)$, where its vertex set is $\mathcal{V}(\mathcal{N}(k)) \subset \mathcal{V}(\mathcal{M})$, and its edge set is $E(\mathcal{N}(k)) = \{e(u, v) | u, v \in \mathcal{V}(\mathcal{N}(k)) \wedge e(u, v) \in \mathcal{E}^k\}$. However, a subnetwork in \mathcal{L}^k can be constructed using the vertex set of community \mathcal{C}_j^i , where $i \leq k$; in which case, the subnetwork is given as: $\mathcal{N}(k, \mathcal{C}_j^i)$, whose vertex set is $\mathcal{V}(\mathcal{C}_j^i)$ and edge set is $E(\mathcal{N}(k, \mathcal{C}_j^i)) = \{e(u, v) | u, v \in \mathcal{V}(\mathcal{C}_j^i) \wedge e(u, v) \in \mathcal{E}^k\}$.

Our proposed focus and context exist in \mathcal{L}^k and pertain to a subnetwork $\mathcal{N}(k)$, and hence, are denoted as $\mathcal{F}(\mathcal{N}(k))$ and $\mathcal{U}(\mathcal{N}(k))$. The shorthand notations for vertex sets of focus and context are V_F and V_U , respectively; and the edge sets are E_F and E_U , respectively. Even though interchangeably used as synonyms, here, we use

¹ The tool is best readable on the Chromium browser.

“network”, “multiplex network”, “nodes” and “links” in the context of dataset, and “graph”, “multigraph”, “vertices”, and “edges” as data structures, respectively.

2 Related Work

In our work, visualizing communities within a SWN and exploring them are key ideas. Prior to visualizing, we detect communities using state-of-the-art algorithms; and for exploring the communities, we use matrix seriation. While there is not much material on visualization of multiplex networks, CD in multiplex networks has been an active area of research. Notwithstanding, as SWNs is a class of complex networks, here, we discuss relevant literature in complex networks as well.

Visualization of Communities in Complex Networks: NodeTrix [13], is a visualization of social networks, where the small-world property of “globally sparse but locally dense” has been exploited to provide the visual representation, which integrates better readability of node-link and matrix representations of the network in respective scenarios (i.e. sparse and dense nature of the network which in the global and local spatial context, respectively) [12]. The locally dense subgraphs are represented as “*aggregated nodes*” (ANs), and rendered as matrices. We direct the readers to the state of the art article on visualizations of groups in graphs [29]. Node-link diagrams and integrated (linked) views have been widely used for visualizing hierarchical structures in networks [25, 26, 30], and for multivariate networks [11, 15, 18]. Bastian et al. [2] have proposed Gephi, a popular network visualization tool, which shows connected components and communities using node-link diagram.

Community Detection in Complex Networks: Modularity-based Louvain CD [7] and graph-theoretic based Tarjan’s algorithms [28] are popularly used for extracting communities and strongly connected components in networks, respectively. Algorithms for hierarchical CD in multiplex networks, for finding crisp communities, use modularity across layers/slices as a guiding principle [5, 19], to determine the best community formation. While these algorithms have composited layers in the multiplex network at the node-level, we propose to perform the same at a coarser level of granularity, i.e. we composite communities, or subnetworks; to make it more scalable for interactive visualizations. de Domenico et al. [10] have proposed the use of modular flows between nodes across layers to identify overlapping communities in multilayer networks. We use a similar concept, except that de Domenico et al. have proposed modular flows across several layers in communities, whereas ours pertain to “modular flows” in aggregated nodes (as used in NodeTrix) across layers in multiplex networks. There have been several studies on visual analytics of multiplex networks such as, Renoust et al. [22] and Rossi and Magnani [24], that have discussed the limitations of extending simplex network visualizations to multiplex ones. They have worked with each network “slice” or layer having its own independent graph layout. As opposed to their work which focuses on visual analytics of dynamics across layers using node-link diagrams predominantly, our work is on CD across layers using a

hybrid visualization. Our visualization is however biased towards the SWN layer, owing to which we do not compute layouts for other layers.

Matrix Seriation: Seriation is a process of reordering rows or columns in a matrix to identify pertinent patterns of clustering. Visual assessment of clustering tendency (VAT) algorithm [6] computes the minimum spanning tree of the dissimilarity graph to give ordering of nodes, and upon reordering, the clusters show the pattern of square blocks along the diagonal of the matrix. Parveen et al. [21] have demonstrated that similarity matrices, after automatic seriation using VAT algorithm, can provide effective matrix visualization of SWNs. We direct the readers to surveys of matrix reordering methods for different domains [17] and for network visualization [4].

3 Focus+Context Approach and Data Model

We propose a focus+context paradigm to probe communities in a subnetwork of interest within the multiplex network. Since we are interested in studying multiple layers of the complex network, our paradigm must be integrated with a HDM. Our rationale is that the focus, which is a subnetwork, will allow us to study localized trends of the network. At the same time, the focus has to be studied in the presence of context, for which we use the rest of the network. In our work, we propose to use a subnetwork ($\mathcal{N}(k)$) in a specific layer (\mathcal{L}^k) as the focus ($\mathcal{F}(\mathcal{N}(k))$); thus, the remaining network becomes the context ($\mathcal{U}(\mathcal{N}(k))$). The vertex and edge sets for the focus (V_F and E_F) and context (V_U and E_U) are:

$$\begin{aligned} V_F &= \mathcal{V}(\mathcal{F}(\mathcal{N}(k))) = \mathcal{V}(\mathcal{N}(k)); \\ E_F &= E(\mathcal{F}(\mathcal{N}(k))) = E(\mathcal{N}(k)) \cup \{e(u, v) \mid (u \in V_F \wedge v \in V_U \wedge e(u, v) \in \mathcal{E}^k) \vee \\ &\quad (u \in V_U \wedge v \in V_F \wedge e(u, v) \in \mathcal{E}^k)\}; \\ V_U &= \mathcal{V}(\mathcal{U}(\mathcal{N}(k))) = \mathcal{V}(\mathcal{M}) \setminus V_F; E_U = E(\mathcal{U}(\mathcal{N}(k))) = \mathcal{E}^k \setminus E_F. \end{aligned} \quad (1)$$

In order to find a *subnetwork of interest*, we propose to perform CD in the concerned layer \mathcal{L}^k , thus getting M_k non-overlapping communities $\mathcal{C}_0^k, \dots, \mathcal{C}_{M_k-1}^k$; and then, use one of the communities as a *subnetwork of interest*. Thus, one such community is treated as the focus, and the remaining network becomes the context. Thus, V_F, E_F, V_U, E_U in Equation 1 can now be written as: $\mathcal{V}(\mathcal{F}(\mathcal{N}(k, \mathcal{C}_j^k))), E(\mathcal{F}(\mathcal{N}(k, \mathcal{C}_j^k))), \mathcal{V}(\mathcal{U}(\mathcal{N}(k, \mathcal{C}_j^k)))$ and $E(\mathcal{U}(\mathcal{N}(k, \mathcal{C}_j^k)))$, respectively.

Using the aforementioned construction of focus, the communities and the focus+context paradigm lie in the same layer, and hence, this pertains to analysis of a single-layer network. What if we use the community in one layer to define the focus, which is further studied across multiple layers in a multiplex network ?

There is a subtle difference between our usage of terms, “community” and “focus”. The edge set of the former consists of the intra-community edges exclusively; whereas that of the latter (E_F , as used in Equation 1) is the set

of all edges (both intra-community edges and inter-community), for which at least one of the vertices belong to the community.

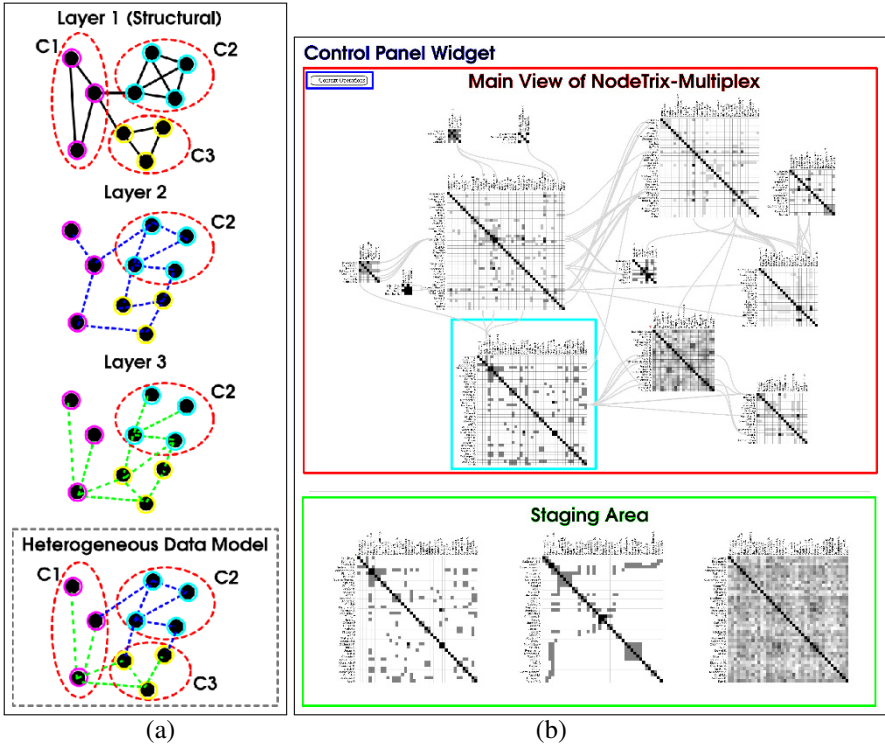


Fig. 1: (a) Schematic of our HDM for multiplex network with three layers, the structural layer (Layer 1) and two functional layers (Layers 2 and 3). Of the communities C1, C2, C3 in Layer 1, the intra- and inter-community edges of the focus (i.e. C2) can be taken from Layer 2 [blue dashed lines]; and those of the context from Layer 3 [green dashed lines]. (b) GUI layout of NTM shows the *main view* [red], widget for expanding the *control panel* [blue] and the *staging area* [green]. A subnetwork of IV dataset (233 nodes, 569 edges, 12 different communities/ANs), with the co-authorship layer in both ANs and links is displayed in the main view. Images of the focus/AN [cyan] from the main view are saved in its staging area; showing (left-to-right) unsorted co-authorship layer, VAT-sorted co-authorship layer, and VAT-sorted author-topic similarity layer.

Heterogeneous Data Model: For a multiplex network, we propose the construction of a *composed single-layer network* \mathcal{M}_{mod} , which is an aggregate of multiple network layers. Our proposed algorithm, of $\mathcal{O}(|\mathcal{V}(\mathcal{M})|)$ complexity, aggregates a

maximum of three layers of \mathcal{M} , taken at a time, in a three-step process (Figure 1(a)). Firstly, we perform CD in layer \mathcal{L}^i to find *subnetwork of interest* \mathcal{C}_j^i . Secondly, using the vertex set $V_F = \mathcal{V}(\mathcal{C}_j^i)$ in layer \mathcal{L}^k we construct *focus*, $\mathcal{F}(\mathcal{N}(k, \mathcal{C}_j^i))$. Thirdly, we define *context*, $\mathcal{U}(\mathcal{N}(u, \mathcal{C}_j^i))$, using vertex set, $V_U = \mathcal{V}(\mathcal{M}) \setminus V_F$, but edge set from a third layer \mathcal{L}^u . Since, we are able to reconstruct a single “*composite*” layer using multiple layers, we call this construction a *heterogeneous data model*. Thus, rewriting Equation 1 for multiple layers:

$$\begin{aligned} E_F &= E(\mathcal{F}(\mathcal{N}(k, \mathcal{C}_j^i))) = E(\mathcal{N}(k)) \cup \{e(u, v) | (u \in V_F \wedge v \in V_U \wedge e(u, v) \in \mathcal{E}^k) \vee \\ &\quad (u \in V_U \wedge v \in V_F \wedge e(u, v) \in \mathcal{E}^k)\}; \\ E_U &= E(\mathcal{U}(\mathcal{N}(u, \mathcal{C}_j^i))) = \mathcal{E}^u \setminus \{e(u, v) | (u \in V_F) \vee (v \in V_F)\} \end{aligned} \quad (2)$$

Our rationale is that we can *switch between different layers* in the focus and context and study *localized patterns*, such as in CD, persistent across the layers.

Since, in our case, the structural layer exhibits the small-world property and contains “*locally dense*” subnetworks, we perform CD in \mathcal{L}^0 . The sparse links between these communities in \mathcal{L}^0 also indicate that the communities internally are well-connected, which implies analysis of each of these communities can be performed mostly independently. Hence, owing to the better defined community formation in \mathcal{L}^0 , our analysis and graph layout are more biased to it than to the other layers. We use one such community in \mathcal{L}^0 as the focus. We find: $V_F = \mathcal{V}(\mathcal{C}_j^0)$; $V_U = \mathcal{V}(\mathcal{M}) \setminus V_F$; $E_F = E(\mathcal{F}(\mathcal{N}(k, \mathcal{C}_j^0)))$; and $E_U = E(\mathcal{U}(\mathcal{N}(u, \mathcal{C}_j^0)))$. This model can be generically used for two-layer multiplex network, where one of the two layers can be treated as \mathcal{L}^u , as done in our case-study.

4 NodeTrix-Multiplex: A Visual Analytic Framework

We propose NodeTrix-Multiplex (NTM), which is a visual analytic framework built on the concepts and visualization layout used in NodeTrix [13]. NTM is a human-in-the-loop framework, which enables users to visually explore and find strong communities which *percolates* across layers of a multiplex SWN. It is integrated with our HDM, which uses focus+context paradigm and a seriation algorithm. It enables the user to understand the dynamics of community formation in different layers by drilling down a subnetwork of interest. The choice of using NodeTrix over node-link diagrams, e.g. in Gephi [2], is due to clear separability of the matrix visualization of focus from the context, in the former (Figure 2). This separability helps in visualization of composited network layer, using different layers for CD, the focus, and the context (Figure 1(a)).

GUI Layout and User Interactions: The proposed layout of GUI for NTM (Figure 1(b)) consists of three components: main view, staging area, and control panel. The hybrid visualization of the focus+context is shown in the **main view**, where the user can choose a focus. The user can interact with the focus and context simultaneously or exclusively with either. In the **staging area** the user can save images

of the focus and view them in different zoom levels. In **the control panel**, the user has the controls to choose the layer for focus/ context visualization, threshold for ϵ -neighborhood for similarity graph (i.e., if a similarity layer is present in the network), color scheme for colormapping of matrices, and seriation. These operations are for the focus and its context, which can be applied simultaneously or exclusively to either, using locking of focus. Separate choices of layer for the focus and the context support the HDM (Section 3) and VAT seriation for the focus (Section 2).

Key Differences between NodeTrix and NTM:

1. NodeTrix is exclusively for studying all ANs in a single-layer SWN homogeneously; whereas our goal is to study local trends in the the multiplex SWN heterogeneously. Our heterogeneous study implies studying an AN in settings different from those of other nodes/ ANs in the network.
2. Owing to the difference in the motivation, NodeTrix uses user-guided agglomeration to create ANs, whereas we use Louvain CD algorithm [7] to automatically extract strong communities in the structural layer. The communities are represented as ANs in NTM.
3. NodeTrix uses user-guided seriation for finding patterns in matrices, whereas we use automatic seriation algorithm, such as VAT algorithm [6].
4. NodeTrix visualizes unweighted adjacency matrix, whereas NTM uses weighted adjacency matrices, for CD, and their complements, i.e. distance matrices, for visualization. The latter is done to comply with the visualization used in VAT algorithm. The difference is that the diagonal cells of AN have value one in NodeTrix (colored white) and value zero in NTM (colored black).
5. The visualization tasks are different – the tasks in both NodeTrix and NTM are to identify communities (T1), central actors (T2), and roles and positions (T3); and NTM additionally has to analyze CD across layers. NTM accomplishes T1 without visual interaction. For T2 and T3, VAT seriation of ANs in NTM highlights the cross, block, and intermediate pattern, as in [13]. The additional unique tasks for NTM are: (T4) find a set of nodes in a community which show clustering tendency across different layers, using the focus, and (T5) find inter-community relationships which could be strong in layers different from the one used for CD, using focus+context.

Figure² 1 shows the layout of the GUI. In the main view, the user can move matrices of the aggregated nodes, which updates the links between the ANs. The operations, which are facilitated through the control panel of NTM, are implemented on both the focus as well as the context. Additionally, depending on the user's needs, these operations can be implemented separately, for which we introduce the notion of

² All images in this paper look best when zoomed in.

“locking” the focus, to preserve it from the modifications made to the context. Thus, the user can choose a focus and activate it, and by locking it, the user activates the context. A blue lock icon in the top left corner of the matrix indicates active or locked state, respectively. A focus can be activated by clicking in the region of the AN. When a focus is deactivated, the user can choose another AN as focus. Extending the layout in NodeTrix to render the focus, we additionally render inter-community links from the AN representing the focus. These inter-community links exist in the layer, which is used for visualizing the focus; while we also render (inter-community) links between ANs in the layer used for visualizing the context.

Software Implementation: NTM has been implemented using Python v2.7 for data preprocessing, Flask framework, and D3.js [9] for visualization. D3.js enables us to perform progressive rendering of sparse links when moving the ANs.

5 Case-Study of a Multiplex Collaboration Network

Our case study, Infovis (IV) co-authorship network [14] during (1995-2015) has 1235 nodes, 2705 edges, 150 communities (detected using Louvain CD). The two layers in IV dataset are co-authorship (structural) and author-topic similarity [23] (functional) graphs. The co-authorship layer (Figure 2) has links between authors if the authors have co-authored, and the edge weight is the number of papers they have co-authored in the topic of Infovis during 1995-2015. The following metadata for each paper is available in the IV dataset: title, authors, keywords, abstract, and references. We have used the metadata to compute the author-topic similarity matrix, which is the adjacency matrix of a similarity graph. Similar to NodeTrix, tasks T2 and T3 can be accomplished from NTM, where the mostly colored row and column (yellow highlights in Figure 3) pertaining to Ben Shneiderman and Jeff Heer, show them to be the central actor in the communities in foci F1 and F2, respectively. Similarly, S. Carpendale, C. North, P. Hanrahan, J. Wood, J. Fekete, J. Dykes, and H. Hauser are central actors in their respective communities/ANs.

NTM helps us find clusters along the diagonal, given by VAT, which recur in multiple layers; e.g. blue, green, and orange highlights in Figure 3 who group together in both the layers. Thus, the staging area (Figure 1(b)) helps in accomplishing task T4. The semantics of such a cluster is that, co-authors in it publish in similar topics, even in papers other than their joint papers. In such clusters in F1 and F2, which also contain the central actors (blue highlights in Figure 3), we observe that the cluster in the structural layer are rendered darker than those in the functional layer, which indicates more accurate similarity scores. On the contrary, the reverse observation in the orange (in F2) and green (in F1 and F2) highlights, where the cluster in the functional layer is darker than its counterpart in the structural layer, indicates erroneous computation of the similarity scores. We have found out that the error in author-topic similarity arises owing to the authors having only one paper in the dataset. Author-topic similarity score is computed using a mixture of distributions associated with the authors in a multi-author paper. A cluster, which is darker in the

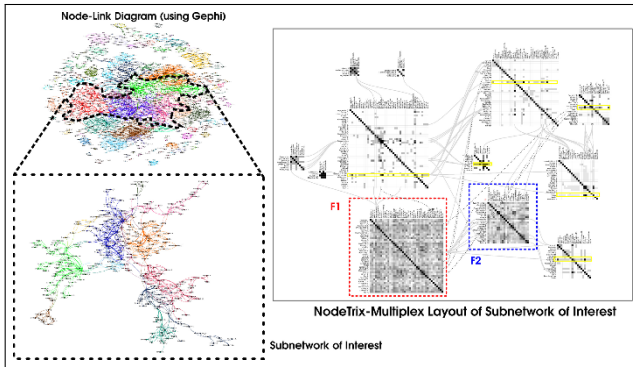


Fig. 2: A subnetwork (233 nodes, 569 edges, and 12 ANs/communities) in the IV co-authorship network dataset shows the foci, F1 and F2, in the author-topic similarity graph (functional layer) and context in the co-authorship layer (structural layer). Yellow highlights show central actors in the community/AN. The inter-community edges are shown in both functional [dotted lines, showing 22 edges with similarity score > 0.7] and structural [solid lines] layers.

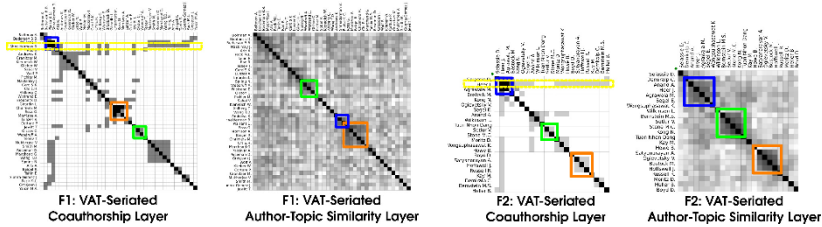


Fig. 3: An aggregated node showing a community in structural layer of IV dataset, after VAT seriation shows clusters recurring in both structural and functional layers [green, blue, orange]. The yellow highlights show central actors.

structural layer than the functional one, implies that the authors have co-authored multiple papers together, owing to which the author-topic similarity scores are more accurate. e.g. {Shneiderman, Plaisant} and {Heer, Agrawala} have authored {8, 4} and {17, 6} independently, and 2 and 5 papers jointly, and thus, have more accurate author-topic similarity scores, 0.57 and 0.60, respectively. Thus, our visualization not only identifies clusters that recur across layers, the aforementioned pattern can help ascertain the accuracy of the results. A corollary to T4 would be to find authors who have not co-authored but have a high author-topic similarity score, which may indicate potential collaboration outside of this network, e.g. {Heer, Stone}³. However, these aforementioned patterns are specific to the current scenario of co-authorship

³ Lin, Sharon, Julie Fortuna, Chinmay Kulkarni, Maureen Stone, and Jeffrey Heer. "Selecting Semantically Resonant Colors for Data Visualization." In Computer Graphics Forum, vol. 32, no. 3pt4, pp. 401-410. Blackwell Publishing Ltd, 2013.

and author-topic similarity layers, and should not be generalized. Nonetheless, NTM enables identification of such trends.

NTM is designed to study all aspects of the subnetwork, corresponding to the focus (which is in structural layer), in all functional layers, for visual analytics; without assuming that the focus remains a community across all functional layers. e.g., links in the similarity layer, but between the AN's in the SWN layer, give more information about the overlap of topics the authors work with, thus accomplishing task T5 (Figure 2). Between ANs with Hauser and Shneiderman as central actors, links $\{\text{Ledermann, Aris}\}$ and $\{\text{Doleisch, Aris}\}$ have been observed to exist due to common topics of *plots* and *user interactions*; and $\{\text{Hauser, Yalcin}\}$, due to the topic of *set visualizations*. Similarly between ANs with Fekete and Shneiderman as central actors, links $\{\text{Henry, Woodruff}\}$ and $\{\text{Ghoniem, Sabol}\}$ have been observed to indicate common topics of *multiple views* and *graph visualization*, respectively.

Work-flow for Community Exploration: Our work-flow for CD and exploration in a multiplex network, using NTM GUI, is a four-step process (Figure 1). Firstly, we input a multiplex network, \mathcal{M} , with N layers, and set the structural layer \mathcal{E}^0 . In our implementation, we construct the multiplex network using author-topic similarity graph, which is the adjacency graph of a functional layer. Similar to NodeTrix [13], NTM becomes slow for interactive response, when the entire network is loaded. For interactive performance, in our case study, we have used Louvain CD ($\mathcal{O}(|\mathcal{V}(\mathcal{M})| \log(|\mathcal{V}(\mathcal{M})|))$) complexity) to identify communities on the structural layer of the entire network, to find logical subnetworks of size upto 250 nodes, to be loaded on NTM. Here, we have used the vertex set of three largest communities in the network as our subnetwork of interest. This step will, however, not be required once NTM is scaled to handle loading of the entire network. Secondly, Louvain CD is performed on the structural layer of the subnetwork, which is loaded on NTM, as a preprocessing step. In our specific case, performing Louvain CD on the entire network and on the subnetwork yield different results; hence, we repeat running the algorithm on the subnetwork after it is loaded. Thirdly, the user can interact with the tool, and pick an AN as a focus. Fourthly, the user can build multiple HDMs, and perform automatic seriation on the AN, using VAT, to visualize possible clusters in each of the layers. For further analysis, different images of the focus are saved and loaded in the staging area.

Expert User Evaluation: We have performed an expert user evaluation of the tool, which is built using NTM as a framework and is available at <http://nmultiplex.au-syd.mybluemix.net/>. The expert, who is a network science researcher, analyzed the usefulness and usability of the tool. The expert mentioned that the use of focus+context visualization helps in focused analysis of communities and hence, the HDM is useful. We have presented the visualizations of the HDM in an existing tool, Gephi (Figure 4), and NTM (Figures 1 and 2), to the

expert. The expert mentioned that the visualizations are better readable on NTM than on Gephi. The expert commented that the HDM and the tool are useful for finding relevant nested communities, which gives a *mesoscopic* network analysis. The ability to switch across different layers allows the user to get an overview of the dynamics occurring in each layer. While the tool does not automate community analysis across the layers, the expert was able to study each focus in detail using the tool. However, the tool is limited in answering specific questions within foci or communities alone, and in its current state, the tool cannot perform a generic analysis of all communities. It also cannot give comparisons of the “strength” of communities across layers. Nevertheless, overall evaluation has been encouraging.

Usability Evaluation: The expert commented that the tool is predominantly easy to use, with the help of the interactive tutorial. The interactivity is responsive, especially due to updates using progressive rendering. The expert liked the color combinations for improving the visual experience. At the same time, the expert pointed out the limitations in the usability of the current version, such as overloading of features on the right mouse button and non-intuitive user interaction for panning in the main view. Currently, the right mouse button is used for selecting focus, popping up the browser menu, and dragging the focus; the scroll wheel is used for zooming in and out; and dragging the left and right mouse buttons has been used for panning. The limitations can be alleviated with UI re-design of the tool.

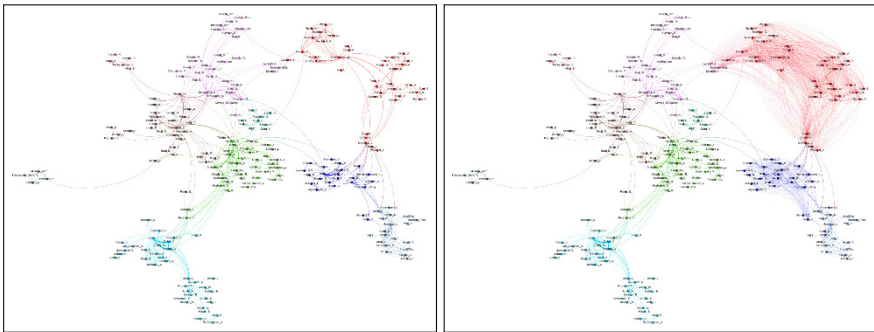


Fig. 4: An equivalent of graph layout in Gephi of the subnetwork of interest, showing the communities in the structural layer, detected using Louvain CD in different colors. Foci F1 and F2 in red and blue, in the structural layer in left, and in similarity layer in the right. The latter shows the node-link diagram of the HDM.

6 Conclusions

We have proposed and implemented a visual analytic framework, NTM, for probing a subnetwork of interest, chosen as a focus, in a multiplex SWN. We have used a focus+context paradigm, our proposed HDM, visual analytic workflow and seriation

for clustering. We have constructed a multiplex network from a co-authorship network (structural layer) by computing author-topic similarity graph as the functional layer. However, there are few limitations in our current approach. In this work, we have focused on multiplex SWNs, owing to which the network topology of the structural layer is restrictive. At the same time, in order to extend this work to different kinds of multiplex networks, without none of the layers exhibiting the small world property, we need to consider an appropriate visual representation of the concerned network topology. NTM, being an extension of NodeTrix, is effective as a hybrid visualization of node-link diagrams and matrix visualization, as the “globally sparse” property of the SWNs reduces clutter and occlusion in the visualization. If the intercommunity links were not to be as sparse as seen in the SWN topology, then the hybrid visualization gets very cluttered. We are currently working on improving scalability in using multiplex networks with more than two layers. We are also investigating other graph layouts, without a bias on SWN layer.

Acknowledgements This work has been partially supported by funding from NRDMS, Department of Science & Technology, Government of India; RSA division of EMC² India; and INCOIS, Ministry of Earth Sciences, Government of India. The authors are grateful to Dr. T. K. Srikanth for his valuable contributions in improving the tool, and to the anonymous reviewers for comments in improving the paper.

References

- [1] Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. *Reviews of modern physics* **74**(1), 47 (2002)
- [2] Bastian, M., Heymann, S., Jacomy, M., et al.: Gephi: an open source software for exploring and manipulating networks. *ICWSM* **8**, 361–362 (2009)
- [3] Battiston, F., Nicosia, V., Latora, V.: Structural measures for multiplex networks. *Physical Review E* **89**(3), 032,804 (2014)
- [4] Behrisch, M., Bach, B., Riche, N.H., Schreck, T., Fekete, J.D.: Matrix reordering methods for table and network visualization. In: *Computer Graphics Forum*, vol. 35, p. 24 (2016)
- [5] Bennett, L., Kittas, A., Muirhead, G., Papageorgiou, L.G., Tsoka, S.: Detection of composite communities in multiplex biological networks. *Scientific reports* **5** (2015)
- [6] Bezdek, J.C., Hathaway, R.J., Huband, J.M.: Visual assessment of clustering tendency for rectangular dissimilarity matrices. *Fuzzy Systems, IEEE Transactions on* **15**(5), 890–903 (2007)
- [7] Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* **2008**(10), P10,008 (2008)
- [8] Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C.I., Gómez-Gardeñes, J., Romance, M., Sendiña-Nadal, I., Wang, Z., Zanin, M.: The structure and dynamics of multilayer networks. *Physics Reports* **544**(1), 1–122 (2014)
- [9] Bostock, M., Ogievetsky, V., Heer, J.: D³ data-driven documents. *IEEE transactions on visualization and computer graphics* **17**(12), 2301–2309 (2011)
- [10] De Domenico, M., Lancichinetti, A., Arenas, A., Rosvall, M.: Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Physical Review X* **5**(1), 011,027 (2015)

- [11] van den Elzen, S., van Wijk, J.J.: Multivariate network exploration and presentation: From detail to overview via selections and aggregations. *Visualization and Computer Graphics, IEEE Transactions on* **20**(12), 2310–2319 (2014)
- [12] Ghoniem, M., Fekete, J.D., Castagliola, P.: A comparison of the readability of graphs using node-link and matrix-based representations. In: *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pp. 17–24. Ieee (2004)
- [13] Henry, N., Fekete, J.D., McGuffin, M.J.: Nodetrix: a hybrid visualization of social networks. *Visualization and Computer Graphics, IEEE Transactions on* **13**(6), 1302–1309 (2007)
- [14] Isenberg, P., Heimerl, F., Koch, S., Isenberg, T., Xu, P., Stolper, C., Sedlmair, M., Chen, J., Möller, T., Stasko, J.: Visualization publication dataset. Dataset: <http://vispubdata.org/> (2015). URL <http://vispubdata.org/>. Published Jun. 2015
- [15] Jusufi, I., Kerren, A., Zimmer, B.: Multivariate Network Exploration with JauntyNets. In: *Proceedings of the 17th International Conference on Information Visualisation (IV'13)*, pp. 19–27. IEEE Computer Society Press (2013)
- [16] Kivela, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. *Journal of complex networks* **2**(3), 203–271 (2014)
- [17] Liiv, I.: Seriation and matrix reordering methods: An historical overview. *Statistical analysis and data mining* **3**(2), 70–91 (2010)
- [18] Martins, R.M., Andery, G.F., Heberle, H., Paulovich, F.V., de Andrade Lopes, A., Pedrini, H., Minghim, R.: Multidimensional projections for visual analysis of social networks. *Journal of Computer Science and Technology* **27**(4), 791–810 (2012)
- [19] Mucha, P.J., Richardson, T., Macon, K., Porter, M.A., Onnela, J.P.: Community structure in time-dependent, multiscale, and multiplex networks. *science* **328**(5980), 876–878 (2010)
- [20] Park, H.J., Friston, K.: Structural and functional brain networks: from connections to cognition. *Science* **342**(6158), 1238,411 (2013)
- [21] Parveen, S., Sreevalsan-Nair, J.: Visualization of small world networks using similarity matrices. In: *Big Data Analytics*, pp. 151–170. Springer (2013)
- [22] Renoust, B., Melançon, G., Munzner, T.: Detangler: Visual analytics for multiplex networks. In: *Computer Graphics Forum*, vol. 34, pp. 321–330. Wiley Online Library (2015)
- [23] Rosen-Zvi, M., Chemudugunta, C., Griffiths, T., Smyth, P., Steyvers, M.: Learning author-topic models from text corpora. *ACM Transactions on Information Systems (TOIS)* **28**(1), 4 (2010)
- [24] Rossi, L., Magnani, M.: Towards effective visual analytics on multiplex and multilayer networks. *Chaos, Solitons & Fractals* **72**, 68–76 (2015)
- [25] Rufiange, S., McGuffin, M.J., Fuhrman, C.P.: Treematrix: A hybrid visualization of compound graphs. In: *Computer Graphics Forum*, vol. 31, pp. 89–101. Wiley Online Library (2012)
- [26] Shi, L., Cao, N., Liu, S., Qian, W., Tan, L., Wang, G., Sun, J., Lin, C.Y.: Himap: Adaptive visualization of large-scale online social networks. In: *Visualization Symposium, 2009. PacificVis' 09. IEEE Pacific*, pp. 41–48. IEEE (2009)
- [27] Shneiderman, B.: The eyes have it: A task by data type taxonomy for information visualizations. In: *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pp. 336–343. IEEE (1996)
- [28] Tarjan, R.: Depth-first search and linear graph algorithms. *SIAM journal on computing* **1**(2), 146–160 (1972)
- [29] Vehlow, C., Beck, F., Weiskopf, D.: The state of the art in visualizing group structures in graphs. In: *Eurographics Conference on Visualization (EuroVis)-STARs*, pp. 21–40 (2015)
- [30] Vehlow, C., Reinhardt, T., Weiskopf, D.: Visualizing fuzzy overlapping communities in networks. *Visualization and Computer Graphics, IEEE Transactions on* **19**(12), 2486–2495 (2013)
- [31] Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. *nature* **393**(6684), 440–442 (1998)

Part VIII
Social and Political Networks

Structural Patterns of the Occupy Movement on Facebook

Michela Del Vicario, Qian Zhang, Alessandro Bessi, Guido Caldarelli and Fabiana Zollo

Abstract The Occupy movement protests against social and economic inequality around the world. It emerged in New York City's Zuccotti Park in September 2011 and is organized at a city level. In this paper we study its social organization on Facebook, by means of a thorough quantitative analysis on users' content consumption. In particular, we focus on structural patterns of users interaction with the movement pages and on the role of local affiliations on the consumption patterns. First, we characterize users' activity finding that passive endorsement (*liking*) is more dominant than active participation to the debate (*commenting*). Then, we label users according to their mobility patterns across pages of the various local communities, finding that online activities are not locally coordinated by geographically close pages. Indeed, pages linked to major US cities, such as New York, Los Angeles, Boston, drive the diffusion of contents online and serve as coordination points for all other pages.

1 Introduction

Social media have been found to foster aggregation of people around shared interests such as political ideas, narratives, and worldviews [4, 5, 18, 30]. Consequently, sociologists and political scientists explored the environment of Internet-based social movements focusing on communication and organization issues [2, 3, 20, 25, 31, 32, 35, 36]. In particular, two of the most studied aspects are a) the *collective framing* i.e., processes that, out of the essential features of the movement's purpose and struggle,

Michela Del Vicario (e-mail: michela.delvicario@imtlucca.it)✉ · Guido Caldarelli (e-mail: guido.caldarelli@imtlucca.it) · Fabiana Zollo (e-mail: fabiana.zollo@imtlucca.it)

IMT School for Advanced Studies, Lucca, Italy

Qian Zhang (e-mail: zhangqian.rach@gmail.com)
Northeastern University, Boston, USA

Alessandro Bessi (e-mail: alessandro.bessi@iusspavia.it)
IUSS Institute for Advanced Study, Pavia, Italy

establish its narratives, language, and imagery [1, 25]; and b) *resources mobilization*, which refers to all those processes exploited by social movements in order to arrange financial, material, and human resources required to sustain their activities in an efficient way [28]. In that regard, an interesting scenario to be explored is the case of online political movements that coordinate and interact through social media [29]. For instance, Twitter played a prominent role in social movements such as the Egyptian revolutionary protests of 2011 [13, 22, 23, 27] and the Arab Spring [24, 26]. The most investigated aspects in this direction concern information flows and the relationship between news media and information sources [24, 27], analysis of tweets' contents [13], impact of media disruption on the dispersion of the protest [23]. Recruitment patterns on Twitter were investigated in [22], where authors reported evidence of social influence and complex contagion.

In this paper we aim at characterizing the shape of online public debate around the Occupy movement on Facebook. The Occupy movement protests against social and economic inequality and relies on online social media for the diffusion of ideas, the recruitment of people, as well as the promotion of the protest [10, 12]. Indeed, social media are responsible for a substantial simplification in the communication and coordination paradigm of the protest's activities. The majority of existing studies about the Occupy movement online was carried on Twitter. For example, in [14] authors focused on the relationship between the geospatial dimension of the social movement's communication network and its resources mobilization, while in [15, 21] researchers analyzed the evolution of the communication activity and of the topics under discussion. Nevertheless, it has to be considered that the most of the online activity during the Arab Spring was done on Facebook (the total accesses to Twitter were just the 1% of the entire population) [34]. Indeed, it has to be noticed that Facebook user base is much bigger than that of Twitter, making the potential reach of posted information much larger. Moreover, Facebook interaction paradigm is particularly appropriate for online social movements, because users can post information without particular limits and, at the same time, can express a positive feedback (*like*), promote information (*share*), and express their opinion (*comment*). Therefore we decide to restrict our attention to Facebook rather than Twitter. Recently, two other studies presented the dynamics of the Occupy movement on Facebook [11, 19], showing that Facebook is mainly used for the recruitment of people and resources to local occupations, the information sharing and story telling, and across-group exchanges.

Since the Occupy movement online presents a geographical diversification of groups, in this work we address geographical patterns behind information diffusion, focusing on the drivers of the inter-pages communication stream. We perform a thorough quantitative analysis of about 620K users on Facebook, taking into account both the role of polarization and homophily in the communication and interaction scheme [6, 7, 16, 37]. We observe a general preference of users towards passive endorsement (*liking*) rather than active participation to the debate (*commenting*, *sharing*). Such a result could be observed for the whole Facebook, because liking requires much less effort than commenting. Nonetheless, in such a context it assumes a peculiar meaning, denoting the tendency of users to express their support to the

Movement, without participate actively. Finally, we show that online activities are not locally coordinated by geographically close pages. Indeed, pages linked to major US cities drive the diffusion of contents online and serve as coordination points for all other pages, that perform a minor activity in the system. Moreover, we find that the pages' total volume of activity, rather than the geographical proximity, is the main driver for the information and users exchange. Our results seem to support those reported in [9], where the author analyzed records of civil unrest of 170 countries during the period 1919–2008 and presented a nonlinear, spatially extended dynamical model, which reflects the spread of civil disorder between geographic regions connected through social and communication networks.

2 Methods

2.1 Data Collection

Using the Facebook Graph API [17], we collected data from 179 Facebook public US pages about the Occupy movement during the time span September 2011-February 2013. For each page we also have a geographical reference located in the US. We defined the space of our investigation using a published list [8] and categorizing Facebook pages according to their contents and their self description. We decided to stop the data collection on February 2013 because the activity on the monitored pages was very low after that point. To the best of our knowledge, the final dataset is the complete set of all Occupy pages active in the US Facebook scenario in the period immediately following the outbreak of the protest on September 2011 to February 2013. A total of 617,563 users is active, in terms of liking activity, on a set of 753,448 posts. The total number of likes (resp. comments and shares) on the downloaded posts is 5,476,444 (resp., 1,280,771 and 108,559).

2.2 Classification of users activity

We make use of a thresholding technique to divide users into *pages-affiliated* groups. Since a like represents a positive feedback, we choose it as a discriminant to identify the affiliation to one page. In particular, we assume the following classification: users having 95% or more of their liking activity on a particular page are said to be *polarized* on that page, while all other users are classified as *not polarized*. Such a thresholding classification detects 97K polarized users and 60K not polarized users out of 157K users. In order to avoid biases in the procedure, this classification is applied only to users who left at least 5 likes on the Occupy corpus.

2.3 Bipartite networks and backbone filter

A bipartite graph is a triple $\mathcal{G} = (A, B, E)$ where $A = \{a_i | i = 1, \dots, n_A\}$ and $B = \{b_j | j = 1, \dots, n_B\}$ are two disjoint sets of vertices, and $E \subseteq A \times B$ is the set of edges – i.e. edges exist only between vertices of the two different sets A and B . The bipartite graph \mathcal{G} is described by the matrix M defined as:

$$M_{ij} = \begin{cases} 1 & \text{if an edge exists between } a_i \text{ and } b_j \\ 0 & \text{otherwise.} \end{cases}$$

The co-occurrence matrices $C^A = MM^T$ and $C^B = M^T M$ count, respectively, the number of common neighbors between two vertices of A or B . C^A is the weighted adjacency matrix of the co-occurrence graph \mathcal{C}^A with vertices on A . Each non-zero element of C^A corresponds to an edge between vertices a_i and a_j with weight P_{ij}^A . The co-occurrence graph \mathcal{C}^B is built in the same way from the co-occurrence matrix C^B .

Let A be the set of the 179 Occupy US pages, B_1 the set of all posts equivalence classes' representatives (representatives posts for short)¹, and B_2 the set of all polarized users active on A ; the *pages-posts* bipartite network is then defined as the triple $\mathcal{G}_1 = (A, B_1, E_1)$, where an edge $e_{ij}^1 \in E_1$ exists if representative post b_j^1 is shared on page a_i^1 , while the *pages-polarized users* bipartite network is defined as the triple $\mathcal{G}_2 = (A, B_2, E_2)$, where an edge $e_{ij}^2 \in E_2$ exists if polarized user b_j^2 is active on page a_i^2 . For our analysis we used two networks derived as co-occurrence networks of the *pages-posts* and the *pages-polarized users* bipartite networks. Considering the co-occurrence matrices C_1^A and C_2^A we get two co-occurrence networks on the vertex set A : the *pages-reshares* (\mathcal{C}_1^A) and the *pages-common users* (\mathcal{C}_2^A) networks. In \mathcal{C}_1^A an edge between two pages exists if at least one representative post is shared on both pages, while in \mathcal{C}_2^A there is a link between two pages if they share at least one user who is polarized in either page.

We apply the *Backbone Extraction*, presented in [33], to the two aforementioned real networks. Such a method applies a thresholding filter based on the local identification of the statistically relevant weight heterogeneities. This kind of approach is able to filter out the backbone of dominant connections in weighted networks with strong disorder, preserving the structural properties and hierarchies at all scales. The discrimination of the right trade-off between the level of network reduction and the amount of relevant information preserved in the new representation involve additional issues. In many cases, the probability distribution $P(x)$ that any given link is carrying a weight x is broadly distributed, spanning several orders of magnitude. Such a problem is addressed by using the aforementioned method presented in [33].

¹ We can consider the equivalence relation of having the same object ID; two posts are equivalent, and hence belong to the same equivalence class, if they have the same object ID.

3 Results and Discussion

3.1 Consumption Patterns

Online social platforms such as Facebook may reach a broader and more diverse audience than traditional media. The Facebook paradigm offers to all users the chance to take actively part in public debates by commenting or sharing pieces of information. Here, our aim is to characterize the actual extent to which online social media foster an open debate around the Occupy movement. We analyze the information consumption patterns of the US Occupy movement in order to first describe the way users interact and get engaged with the movement online and hence understand the effectiveness of the online media in fostering and shaping the debate. In addition, we characterize the role of polarized users in promoting and bridging inter-page connections. Since pages recall the geolocation of the community we also associate each user to that local affiliation.

As a first step we look at the distributions of the different activities, i.e., the number of posts, likes, comments, and shares, and of the different users' categories, i.e., the number of users, polarized users, and not polarized users, on all the city related pages. Fig. 1 shows the Pearson correlation coefficient for all the distributions pairwise. Notice that they all exhibit high positive correlations, indicating that pages showing a strong commitment (*number of posts*) emerge as hubs connecting like-minded individuals (*number of users*) who endorse, debate, and share information.

We then analyze the information consumption patterns in order to characterize the nature of the online debate around the Occupy movement. It is important to note that, while a like stands for a passive endorsement of the content, a comment denotes an active participation of the user in the debate and a share reflects the will to attract the attention on the post. Left panel of Fig. 2 shows the complementary cumulative distribution functions (CCDFs) of the number of likes, comments, and shares of all the posts of the corpus, while right panel shows the CCDFs of liking and commenting activity of all users. We take into account different fits for the distributions in Fig. 2 (the exponential, the power law, and the lognormal) and we use the *Nonlinear least square estimation (NLS)* to fit them. Goodness of fit tests based on the log-likelihood

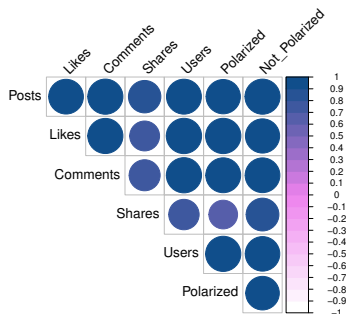


Fig. 1 Pearson correlation for number of posts, likes, comments, shares, total users, polarized users, and not polarized ones is high for all the different combinations.

proved that while the distribution of the number of likes and comments on all posts are exponentially distributed, all the other distributions are better fitted by a power law.

We notice that users have a preference towards likes rather than comments or shares, denoting the tendency to avoid an active participation in the debate around the Occupy movement. In the specific, the probability of diffusing a piece of information, by sharing the relative post more than a given number of times x , shows a drastic decrease for $x > 1$. This result points out that the Facebook pages relative to the Occupy movement mainly serve as a promotional space, while there is no trace of a public debate on them.

3.2 Activity of Polarized Users

In this section, we analyze the information consumption patterns inside the Occupy movement by focusing on the users' activity and comparing the consumption patterns of polarized and not polarized users. In particular, we focus on the difference between the liking and commenting activity across both categories of users (Fig. 3). We observe that polarized users tend to increase the probability of commenting more than x times (for $x \sim 10^3$) rather than just linking, while not polarized users maintain a higher probability of liking at all scales.

We then looked at the users' lifetime, where the lifetime of a user is defined as the temporal distance (counted in days) between her first and last comment on a post of the Occupy movement pages, and in particular at the liking activity of polarized users as a function of lifetime. Fig. 4 shows the CCDF of the number of likes of polarized users for different levels of lifetime.

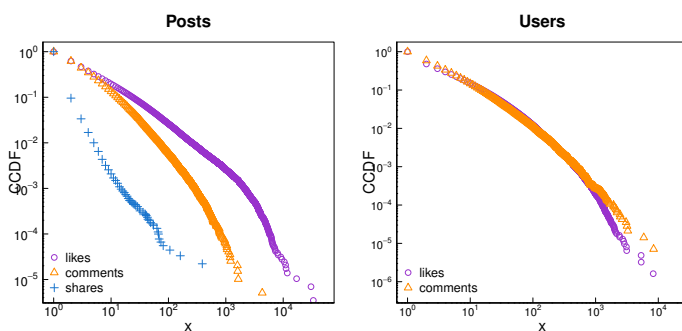


Fig. 2: *Left Panel*: CCDF of the number of likes (violet), comments (orange), and shares (blue) on all posts from the 179 different Occupy pages. Users tend to avoid an active participation in the debate and prefer the passive endorsement of the posts (like). *Right Panel*: CCDF of liking (violet) and commenting (orange) activity of all users.

We notice that there is no significant difference between the CCDFs of the number of likes left by polarized users of varying lifetimes, a higher lifetime is not a synonymous of a higher activity. We pairwise compared the five distributions, i.e., the number of likes of polarized users for a lifetime of, respectively, 1, 10, 100, 200, and 500 days, by Kolmogorov-Smirnov test with the null hypothesis of equivalence of the whole sample distributions and significance level $\alpha = 0.05$. The estimated maximum distance, reported in Table 1, is always smaller than the corresponding critical value and hence we may deduce that there is no significant statistical difference among all five distributions.

Table 1: Estimated maximum distances, with corresponding critical values in brackets, from the Kolmogorov-Smirnov test pairwise applied to the five distributions of number of likes left by polarized users for different levels of lifetime (1 day, 10 days, 100 days, 200 days, and 500 days).

	1 day	10 days	100 days	200 days
10 days	0.062 (0.091)	-	-	-
100 days	0.054 (0.129)	0.056 (0.144)	-	-
200 days	0.108 (0.165)	0.107 (0.177)	0.098 (0.199)	-
500 days	0.069 (0.188)	0.053 (0.199)	0.070 (0.219)	0.116 (0.242)

Dividing the users into different categories, we observe a differentiation in the consumption patterns. Polarized users show an increase in the probability of commenting a post with respect to that of just liking it, however this probability is still

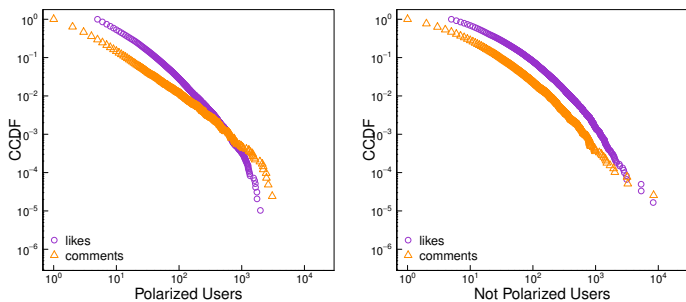


Fig. 3: CCDF of liking (violet) and commenting (orange) activity for polarized (left) and not polarized users (right). By means of NLS estimation and goodness of fit tests based of the log-likelihood, we find that all the distributions are exponentially distributed.

smaller for not very active users. We also find that the liking activity of polarized users is not affected by their lifetime.

3.3 Backbone of Interaction Patterns

One of the most peculiar characteristics of the Occupy movement online is its city level organization. The movement got started in New York and then spread all over the US. We are interested in analyzing the spreading of the movement online and identifying the drivers of the information flow. We consider the diffusion of the information in the system in terms of common shared posts and common polarized users between two pages and we test the geographical proximity and the total number of posts as possible drivers of the diffusion.

In order to discriminate if geographical proximity does actually affect the diffusion of information, we consider two different weighed networks of the 179 geolocated Occupy pages: the *pages-reshares* network and the *pages-common users* network. In the pages-reshares network a link between two pages exists if they shared at least once the same post, while in the pages-common users network a link between two pages exists if they share at least one user that is polarized in either page. Refer to Section *Materials and Methods* for further details on the structure of the two networks. We then apply the *Backbone Extraction Algorithm* [33] to the two networks described above, since this method allows us to filter out the backbone of dominant connections in a weighted network while preserving the structural properties. Fig. 5 illustrates the multi-scale backbone structure for pages-reshares network (*top*), and for the pages-common users network (*bottom*), for the two levels of significance $\alpha = \{0.01, 0.05\}$ (respectively in blue and orange). The links correspond to the

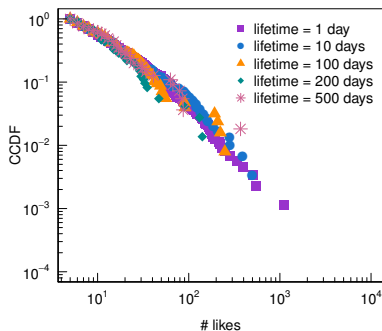


Fig. 4: CCDF of number of likes left by polarized users for different levels of lifetime: violet is for one day lifetime, blue for 10 days, orange for 100 days, green for 200 days, and pink for 500 days.

activity of users polarized on one page inside another page. These results provide a clear image of the absence of geographical correlation in the resharing patterns. Moreover, pages corresponding to the major US cities² emerge as leaders in the information spreading and show an exchange of polarized users' activity. Hence, hubs corresponding the US major cities drive the overall activity of the movement and the diffusion of contents online, serving as coordination points for all other pages.

While there is no correlation between the number of common shared posts (or the number of common users) and the geographical proximity of the pages, there is a positive linear correlation between the number of common shared posts (or the number of common users) and the number of posts on the pages, ~ 0.73 (or ~ 0.89). Fig. 6 shows two chord diagrams where the links in the left one represent the number of common shared posts between the top 6 pages for number of posts, i.e., those pages geolocated in Wall Street, Boston, Portland, Chicago, Denver, and Los Angeles, while the links in the right one represent the number of common users between the same 6 pages. The thicker the link, the higher the common shared posts (or the common users) similarity per number of posts between two cities. We found that the inter-pages communication is driven by the pages' volume of total activity rather than by their geographical proximity.

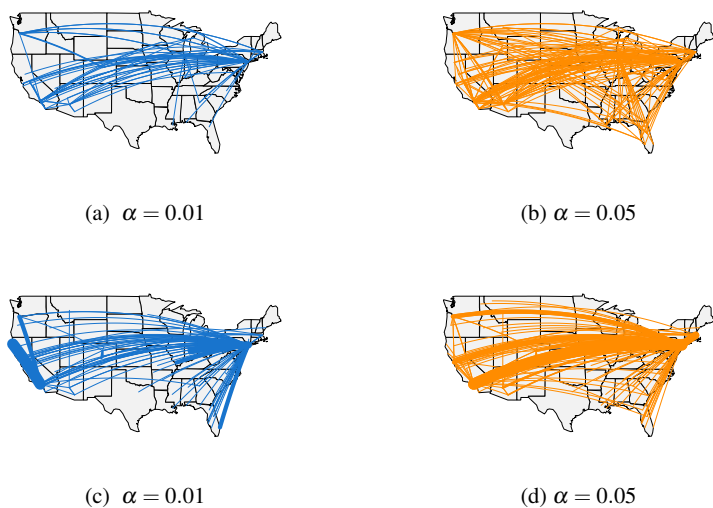


Fig. 5: Backbone structure for the **pages-reshares** network (*top*) and for the **pages-common users** network (*bottom*) for $\alpha = \{0.01, 0.05\}$ (respectively in blue and orange).

² For both levels of significance, the following cities emerge as information spreading leaders: New York, Los Angeles, Chicago, Boston, Portland, Phoenix, and Denver.

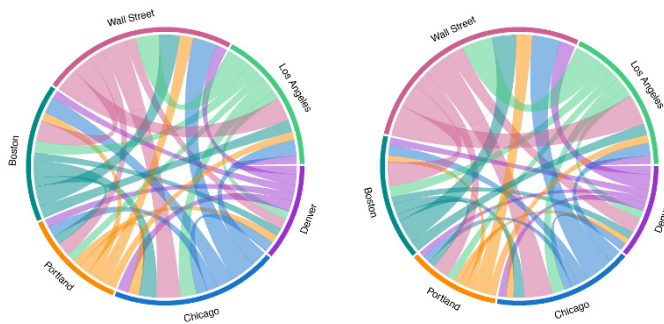


Fig. 6: Chord diagram representing the number of common reshared posts (left) and the number of common users (right) between the top 6 pages for number of posts published, i.e., New York, Los Angeles, Chicago, Boston, Portland, Phoenix, and Denver. The inter-pages communication stream is driven by the pages' volume of total activity.

4 Conclusions

In this paper we explore the case of online political movements that coordinate and interact through social media always more often with the advent of the World Wide Web [29]. Our focus is to characterize the shape of online public debate around the Occupy movement by addressing the information consumption patterns and by identifying different actors according to their interaction patterns with pages and posts. Since the Occupy movement online presents a geographical diversification of groups, we address geographical patterns behind information diffusion and in particular the drivers of inter-pages communication's stream. By analyzing users activity on pages and posts, we find a clear predominance of the likes, with respect to the comments or shares, and we consider this fact as the first evidence of the tendency of users to avoid an active engagement in the debates around the Occupy movement. Also, we notice a differentiation in the consumption patterns of polarized and not polarized users. Indeed, the first ones show an increase in the probability of commenting a post rather than just liking it for high activity levels. This result points out that the Facebook pages relative to the Occupy movement mainly serve as a promotional space, while there is no trace of a public debate on them. Furthermore, we characterize polarized users attitude towards the Occupy movement's online debate by looking at their liking activity as a function of the lifetime, finding that the activity intensity is not affected by the user's lifetime. Moreover, we extract the multi-scale backbone for two networks, the pages-reshares network and the pages-common users network, in order to analyze geographical patterns in the information diffusion and polarized users activity. Our analysis reveals that activities online are not locally coordinated by geographically close pages. Indeed, pages linked to major US cities drive the diffusion of contents online and serve as coordination points for all other pages, which perform a minor activity in the system. We also find a high and

positive linear correlation between the number of common shared posts (or common users) and the number of total posts on the page, that leads to the emergence of the pages' volume of total activity as the inter-pages communication's driver.

Summarizing, in this paper we show that few pages exhibiting a strong commitment (*number of published posts*) emerge as hubs connecting like-minded individuals. The vast majority of both users and information flows between hubs linked to major US cities, whereas we find no evidence of significant flows between pages linked to cities which are geographically close. This, together with the fact that users prefer passive (*likes*) over active (*comments*) endorsements, suggests that Facebook pages related to the Occupy movement mainly served as *virtual plazas* used to raise awareness and promote insurgent beliefs rather than to organize local protests and facilitate the debate around matters of interest. Future works may be devoted to extend the discussion by analyzing page contents and topics. Indeed, understanding how people interact and discuss about the Occupy case, taking also into account their emotional behavior, could provide interesting insights to better define users' response to protest movements online.

Acknowledgements Funding for this work was provided by EU FET project MULTIPLEX nr. 317532, SIMPOL nr. 610704, the FET project DOLFINS 640772 (H2020), SoBigData 654024 (H2020), and CoeGSS 676547 (H2020). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

- [1] Benford, R.D., Snow, D.A.: Framing processes and social movements: An overview and assessment. *Annual review of sociology* pp. 611–639 (2000)
- [2] Bennett, W.: Communicating global activism. *Information, Communication & Society* **6**(2), 143–168 (2003)
- [3] Bennett, W.L.: Changing citizenship in the digital age. *Civic life online: Learning how digital media can engage youth* **1**, 1–24 (2008)
- [4] Bessi, A., Coletto, M., Davidescu, G.A., Scala, A., Caldarelli, G., Quattrociocchi, W.: Science vs Conspiracy: collective narratives in the age of misinformation. *PLoS ONE* **10**(2), 02 (2015)
- [5] Bessi, A., Petroni, F., Del Vicario, M., Zollo, F., Anagnostopoulos, A., Scala, A., Caldarelli, G., Quattrociocchi, W.: Viral misinformation: The role of homophily and polarization. In: *ACM Proceedings of the 24th International Conference on World Wide Web* (2015)
- [6] Bessi, A., Scala, A., Rossi, L., Zhang, Q., Quattrociocchi, W.: The economy of attention in the age of (mis) information. *Journal of Trust Management* (2014)
- [7] Bessi, A., Zollo, F., Del Vicario, M., Scala, A., Caldarelli, G., Quattrociocchi, W.: Trend of narratives in the age of misinformation. *PLoS ONE* **10**(8) (2015)
- [8] Bowers, C.: Occupy Wall Street: List and map of over 200 u.s. solidarity events and facebook pages. Website (2013). URL <http://www.dailykos.com/story/2011/10/04/1022722/-OccupyWall-Street:-List-and-Map>. Last checked: 02.2013
- [9] Braha, D.: Global civil unrest: contagion, self-organization, and prediction. *PLoS ONE* **7**(10) (2012)
- [10] Byrne, J.: Occupy the media: Journalism for (and by) the 99 percent. *The Occupy Handbook*. Little, Brown pp. 256–264 (2012)
- [11] Caren, N., Gaby, S.: Occupy online: Facebook and the spread of occupy wall street. *Social Science Research Network Working Paper Series* (2011)
- [12] Chomsky, N.: Occupy. *Occupied Media Pamphlet Series* **1** (2012)

- [13] Choudhary, A., Hendrix, W., Lee, K., Palsetia, D., Liao, W.K.: Social media evolution of the egyptian revolution. *Communications of the ACM* **55**(5), 74–80 (2012)
- [14] Conover, M.D., Davis, C., Ferrara, E., McKelvey, K., Menczer, F., Flammini, A.: The geospatial characteristics of a social movement communication network. *PLoS ONE* **8**(3) (2013)
- [15] Conover, M.D., Ferrara, E., Menczer, F., Flammini, A.: The digital evolution of Occupy Wall Street. *PLoS ONE* **8**(5), e64,679 (2013)
- [16] Del Vicario, M., Bessi, A., Zollo, F., Petroni, F., Scala, A., Caldarelli, G., Stanley, H.E., Quattrociocchi, W.: The spreading of misinformation online. *Proceedings of the National Academy of Sciences* **113**(3), 554–559 (2016)
- [17] Facebook: Using the Graph API. Website (2013). URL <https://developers.facebook.com/docs/graph-api/using-graph-api/>. Last checked: 19.01.2014
- [18] Friggeri, A., Adamic, L.A., Eckles, D., Cheng, J.: Rumor cascades. *ICWSM* (2014)
- [19] Gaby, S., Caren, N.: Occupy online: How cute old men and malcolm x recruited 400,000 us users to ows on facebook. *Social Movement Studies* **11**(3-4), 367–374 (2012)
- [20] Gaffney, D.: #iranelection: quantifying online activism. *Web Science Conf.* (2010)
- [21] Gargiulo, F., Bindi, J., Apolloni, A.: The topology of a discussion: the #Occupy case. *PLoS ONE* **10**(9), e0137,191 (2015)
- [22] González-Bailón, S., Borge-Holthoefer, J., Rivero, A., Moreno, Y.: The dynamics of protest recruitment through an online network. *Scientific reports* **1** (2011)
- [23] Hassanpour, N.: Media disruption exacerbates revolutionary unrest: Evidence from mubarak’s natural experiment. *American Political Science Association Annual Meeting Paper* (2011)
- [24] Howard, P.N., Duffy, A., Freelon, D., Hussain, M., Mari, W., Mazaid, M.: Opening closed regimes: what was the role of social media during the Arab Spring? Working Paper, PIPTI (2011)
- [25] Kelly Garrett, R.: Protest in an information society: a review of literature on social movements and new icts. *Information, Communication and Society* **9**(2), 202–224 (2006)
- [26] Khondker, H.H.: Role of the new media in the arab spring. *Globalizations* **8**(5), 675–679 (2011)
- [27] Lotan, G., Graeff, E., Ananny, M., Gaffney, D., Pearce, I., Boyd, D.: The revolutions were tweeted: Information flows during the 2011 tunisian and egyptian revolutions. *International Journal of Communications* **5**, 1375–1405 (2011)
- [28] McCarthy, J.D., Zald, M.N.: Resource mobilization and social movements: A partial theory. *American journal of sociology* pp. 1212–1241 (1977)
- [29] McCaughey, M., Ayers, M.D.: *Cyberactivism: Online activism in theory and practice*. Psychology Press (2003)
- [30] Mocanu, D., Rossi, L., Zhang, Q., Karsai, M., Quattrociocchi, W.: Collective attention in the age of (mis) information. *Computers in Human Behavior*, Elsevier (2014)
- [31] Myers, D.J.: Communication technology and social movements: Contributions of computer networks to activism. *Social Science Computer Review* **12**(2), 250–260 (1994)
- [32] Myers, D.J.: Media, communication technology, and protest waves. In: *Trabalho apresentado na conferência Social Movement Analysis: The Network Perspective*, Ross Priory, Scotland, pp. 22–25 (2000)
- [33] Serrano, M.Á., Boguñá, M., Vespignani, A.: Extracting the multiscale backbone of complex weighted networks. *Proceedings of the national academy of sciences* **106**(16), 6483–6488 (2009)
- [34] Tufekci, Z., Wilson, C.: Social media and the decision to participate in political protest: Observations from Tahrir square. *Journal of Communication* **62**(2) (2012)
- [35] Van Laer, J., Van Aelst, P.: Cyber-protest and civil society: the internet and action repertoires in social movements. *Handbook on internet crime* pp. 230–254 (2009)
- [36] Wray, S.: On electronic civil disobedience. *Peace Review* **11**(1), 107–111 (1999)
- [37] Zollo, F., Novak, P.K., Del Vicario, M., Bessi, A., Mozetic, I., Scala, A., Caldarelli, G., Quattrociocchi, W.: Emotional dynamics in the age of misinformation. *PLoS ONE* **10**(9) (2015)

Political Participation in Mexico through Twitter

Julio César Amador Díaz López and C. A. Piña-García

Abstract We used survey data and collected data from the Online Social Network (OSN) Twitter between October the 5th and November the 9th (time window) to provide an overview related to political participation in Mexico. With the survey data we provided a qualitative assessment of political participation in Mexico by examining electoral participation, levels of political participation between regions, Mexicans' interest in politics and their sources of political information. With our collected data, we described the intensity of political participation in this OSN, we identified locations of high Twitter activity and identified political movements including agencies behind them. With this information, we compare and contrast political participation in Mexico to its counterpart through Twitter. We show that political participation in Mexico seems to be decreasing. However, according to our preliminary results political participation in Mexico through Twitter seems to be increasing. In this regard, our research points towards the emergence of Twitter as a significant platform in terms of political participation in Mexico. Our study analyses the impact of how different agencies related to social movements can enhance political participation through Twitter. We show that emergent topics related to political participation in Mexico are important because they could help to explore how politics becomes of public interest. The study also offers some important insights for studying the type of political content that users are more likely to tweet.

1 Introduction

Academics generally agree that political participation is quintessential for democracy. Not only does political participation serve as the main conduit with which the public

Julio César Amador Díaz López (e-mail: j.amador@imperial.ac.uk)
Imperial Business School, Imperial College London

C. A. Piña-García (e-mail: carlos.pgarcia@iimas.unam.mx)
Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, Mexico City

expresses their opinion, but it also establishes an important link between the public, the state and its institutions. In spite of its importance, there is a view amongst researchers suggesting that political participation has decreased in recent decades. The lack of participation is visible through the decrease in turnout during election periods, the growth of negative sentiments towards politicians and their parties, and the decline in engagement in civic associations [6].

Even if political participation seems to be decreasing, the emergence of new agencies, such as online social networking sites, pose the possibility that political participation is shifting away from traditional practices and moving towards online ones. In fact, as the use of online social networks has become mainstream, their role as agents for social change has increased [3]. The use of online social networks in events ranging from the spread of political news, election campaigns and protests has demonstrated their importance in the context of fomenting social change.

Twitter can be considered the most studied online social network. This social media platform provides an efficient and effective communication medium for one-on-one interactions and broadcast calls (e.g., for assistance or dissemination and access to useful information). In Twitter users post messages that are limited to 140 characters known as tweets and it produces around 500 million tweets per day and has 271 million regular users [2].

Because Twitter is used to share information, opinions, and online petitions, the social network provides us with an important source of data useful to analyse online political participation in Mexico. With this in mind, our aim is to briefly assess political participation in Mexico and, through the case study of our data in the OSN Twitter, put forward possible paths for how the emergence of new technologies could enhance political participation. In our context, we refer to political participation as any activity through which individual express their own opinion with the goal of exerting influence regarding political decision-making.

In order to do so, we use survey data to qualitatively assess political participation, interest in politics and sources of political information in Mexico. We present descriptive visualizations on electoral turnout and regions with the largest level of participation. Moreover, we provide trends on the most important sources of political information in Mexico and forms of political participation other than voting. Next, we use a corpus of over 150,000 tweets (dataset) related to the president of Mexico, Enrique Peña Nieto. Additionally, we present graphics that will give us an idea of how frequently people participate with political content in Twitter, their locations and what sort of conversations they have on Twitter. Our data visualizations will allow us to identify different online protests in the Mexican territory.

Our work is related to the research carried out by [1] on political attitudes and civic culture, [6] on political participation in Mexico, to [11] study on organized civil society and democracy in Mexico and a recent study led by [10] on the status of political participation in Mexico.

[1] and [6] provide a starting point for our research through a description of the state of political participation in Mexico. This is, then, updated by [6] which provides the first clues towards the state of political participation. We use these authors views on the state of political participation as a benchmark to what we will find in survey

and Twitter data. As such, our paper provides an update to the state of political participation in Mexico and, to the best of our knowledge, adds the first comparison to online participation through Twitter in Mexico. Moreover, our paper identifies promoters of online political participation online. This complements [11] view of civil associations by highlighting the role of Change.org as promoter of political participation online.

2 Methodology

Given that our aim is to describe political participation in Mexico, we turn to survey and collect data from Twitter to: firstly, explore and examine trends in traditional ways of political participation. Secondly, to spot shifts in sources of political information. Lastly, to assess and investigate trends in non-traditional ways of political participation. It is important to recall that we refer to political participation as any activity through which individual express their own opinion with the goal of exerting influence regarding political decision-making.

2.1 Survey Data

We use two different surveys to gauge political participation in Mexico. On the one hand, we use Election Day turnout data first, from the Federal Electoral Institute, or IFE, and then from the National Electoral Institute or INE [10]. From these sources we obtain turnout for parliamentary and presidential elections between 1964 and 2015. Moreover, we obtain Election Day turnout for each of the states in Mexico between 1991 and 2009. On the other hand, we use data from the National Survey on Political Culture and Citizenship, or ENCUP http://www.encup.gob.mx/en/Encup/Bases_de_datos. This survey examines characteristics and practices of political culture in Mexico. This survey also consists on a National representative sample of the population in Mexico of ages 18 and older. Data is available for the years 2001, 2003, 2005 2008 and 2012 (http://www.encup.gob.mx/en/Encup/Bases_de_datos). In relation to ENCUP 2016, we begin by examining traditional measures of political participation such as attending political meetings, signing letters, calling authorities, participation in civil associations, contacting representatives and political parties and, where available the use of Internet and online social media to access political information. Next, we inquire on the sources of political information of the population. Finally, for the years where present, we describe political participation in non-traditional ways such as online activism and social networks. To gauge political participation in Mexico, we use data from the National Survey on Political Culture and Citizenship, or ENCUP http://www.encup.gob.mx/en/Encup/Bases_de_datos. This survey examines characteristics and practices of political culture in Mexico. This survey also consists on a National representative sample of the population in Mexico of ages 18 and older. Data is available for the years 2001, 2003, 2005 2008 and 2012

(http://www.encup.gob.mx/en/Encup/Bases_de_datos). In relation to ENCUP 2016, we begin by examining traditional measures of political participation such as attending political meetings, signing letters, calling authorities, participation in civil associations, contacting representatives and political parties and, where available the use of Internet and online social media to access political information. Next, we inquire on the sources of political information of the population. Finally, for the years where present, we describe political participation in non-traditional ways such as online activism and social networks.

2.2 *Twitter Data*

Given the nature of this study, it is worth briefly discussing the ethical, legal, and social implications of using Twitter data to conduct research. The tweets that were collected through the public Twitter API are subject to the Twitter terms and conditions. Thus, the privacy policy used by Twitter indicates that users consent to the collection, transfer, manipulation, storage, and disclosure of data are public. This study analyzed only tweets that were completely public (i.e., no privacy settings were selected by the user). Thus, there was no expectation of privacy by the user [5].

We collected publicly available tweets related to the president of Mexico, Enrique Peña Nieto, and his personal Twitter username, also known as Twitter handle, @EPN. The data was collected from October to November 2015 via the Twitter streaming API (<https://dev.twitter.com/streaming/overview>). In this regard, we decided to build a corpus of tweets based on a personalized list of thirteen topics where some of them are paired with Peña Nieto's keyword. This list is as follows: 1.-Peña Nieto, 2.-crisis, 3.-México, 4.-#Ayotzinapa, 5.-corrupción (corruption), 6.-sociedad (society), 7.-derechos humanos (human rights), 8.-periodistas (journalists), 9.-economía (economics), 10.-renuncia (renounce), 11.-petróleo (oil) and 12.-inflación (inflation). This list was identified through empirical consultation and by experimentally querying the Twitter database to investigate which terms were most commonly used. We can also argue that our list of 12 keywords is highly related to our own research interests and we are only interested in tweets posted in Spanish. Therefore, it is important to bear in mind a possible bias in the chosen topics.

These tweets are openly available to the public on the web which implies that protected tweets will not be picked up. Consequently, their use for research is typically thought not to raise any ethical concerns. Twitter provides a continuous stream of public information. It does so by allowing millions of people to broadcast short messages known as “tweets”. In this context, people can “follow” others to receive their messages, forward or “retweet” (“RT” in short) tweets to their own followers, or mention (“@” in short) others in tweets. People often label tweets with topical keywords or “hashtags”. A hashtag is a convention among Twitter users to create and follow a thread of discussion by prefixing a word with a # character. Thus, Twitter tracks phrases, words, and hashtags that are most often mentioned and regularly post them under the title of trending topics.

Our sample consisted of 150,000 tweets published by 46 399 users that emerged during the observed time window. This sample was stored for further analysis. This data collection contains information such as: user ID, date and time that the user account was created, the screen name or alias, the number of followers, time when a tweet was posted, the tweet itself, language, device used to post the tweet (source), and the user-defined location (when tweet location service was on). It is important to note that approximately 1% of all tweets published on Twitter are geo-located. This is a very small portion of the tweets, and it is often necessary to use the profile information to determine the tweets location [4].

Tweets are the basic atomic building block of all things in Twitter. Given that the text of the status update (the tweet itself) includes embedded data such as: retweet, hashtags and mentions, we extracted these four features to improve our exploratory approach.

3 Results

The political community thinks political participation goes beyond the election of representatives [6]. Activities such as protests, joining civil associations and writing letters may be also considered forms of participation. Thus, [6] finds that the levels of this form political participation are, on average, larger in Mexico than in similar countries of Latin America.

The emergence of new ways of participation such as online social networks suggests that agencies through which political participation is channelled might be changing. It is sufficient to notice that, even if the ENCUP http://www.encup.gob.mx/en/Encup/Bases_de_datos started tracking participation in online social networks in 2012, the level of participation in online social networks is comparable to the average level of participation of people calling the radio. In order to single-out changes in the ways new outlets affect social participation, we plot in Fig. 1 trends of the participation agencies discussed above.

Notice that, even if channels of participation such as sending letters, writing to the President and calling the radio appear to lose ground after 2008, participation levels through different channels seem to follow the same trend. Importantly, there appears to be a general decrease in political participation between 2005 and 2008, which is followed by a slight increase between 2008 and 2012.

It is important to note that ENCUP http://www.encup.gob.mx/en/Encup/Bases_de_datos began reporting Internet as a source of information only from 2008 and online social networks on 2012. Hence, Fig. 2 shows trends of the usage of different media as sources of political information.

Even though trends in the usage of different media as sources of political information appear to be the same, the decrease in the use of the Internet as a source of information appears to be less steep than the decrease of other sources of information. However, the television continues to be the largest source of political information followed by the radio.

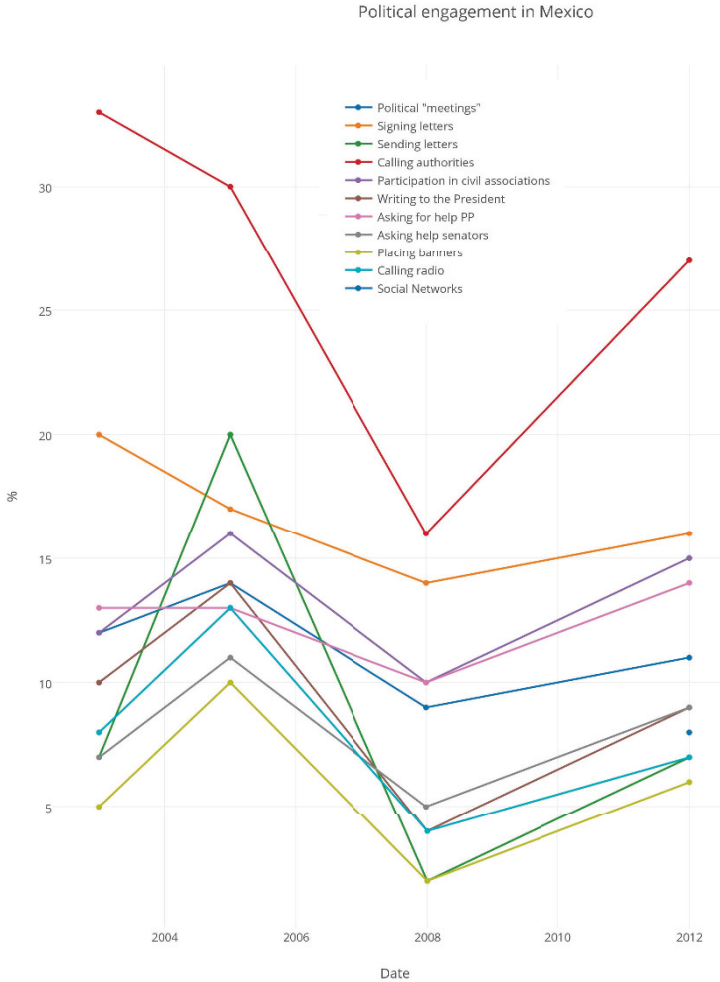


Fig. 1: Political Participation Levels in Non-Electoral Activities between 2001 and 2012.

It is interesting to note that by 2012, the same proportion of the population reported newspapers and the Internet as their main source of political information. Moreover, it should be noted that the proportion of people that reports online social networks as their source of political information is close to those that said comments were their main source of information.

The trends showed above seem to suggest several general patterns. First, that in Mexico there are low levels of interest in politics which correlate and, maybe, translate into a decrease in the levels of political participation in the last 20 years. Second, that even though Mexicans participate politically in activities other than

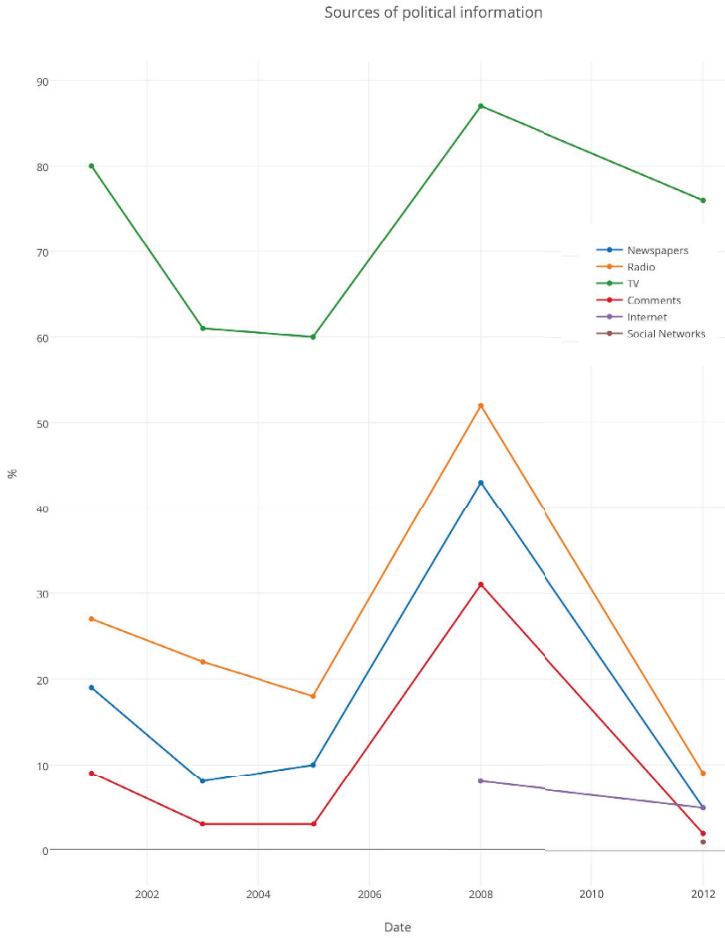


Fig. 2: Sources of Political Information between 2001 and 2012.

voting, such activities appear to be limited in their level of social and political engagement. Third, that in recent years the sources of political information that Mexicans have access to have been disrupted by the emergence of the Internet and online social networks. An example of this is the online social protest #YoSoy132. This protest was mainly organized by university students and began as opposition to the now president of Mexico, then candidate of the Institutional Revolutionary Party or PRI for its acronym in Spanish, and the alleged biased coverage the mainstream media in Mexico had of the 2012 general election.

Following the structure of the analysis above, we begin by looking at the level of political participation in Twitter. In order to gauge online interest in politics, a possible approach is to visualize the daily frequency of tweets related to Enrique

Peña Nieto. As Twitter has become a valuable tool to track and to identify patterns of mobility and activity, we now turn to examine and capture all locations where our collected tweets were posted. According to [4], approximately 1% of all tweets published on Twitter are geo-located i.e., users can optionally choose to provide location information for the tweets they publish. This is a very small portion of the Tweets, therefore, we decided to use the profile information with the aim to determine the location. Fig. 3 provides a geographical heat map mentioning the president of Mexico Enrique Peña Nieto aimed to identify regions of high density of tweeting activity. In this case, the colour scheme denotes blue color to indicate low activity, and red color to indicate high density. Thus, it is possible to see that the central region of Mexico is the one with the highest level of political engagement.

#YoSoy132 highlights the importance of the OSN Twitter in the diffusion of political information in Mexico. As such, we now move our attention to the case presented by our Twitter data. It is important to stress that the analysis that follows is limited to our sample and by no means intends to generalize our findings to political participation in Mexico. However, for ease of exposition, we will refer to political participation levels in Twitter within our sample simply as political participation.

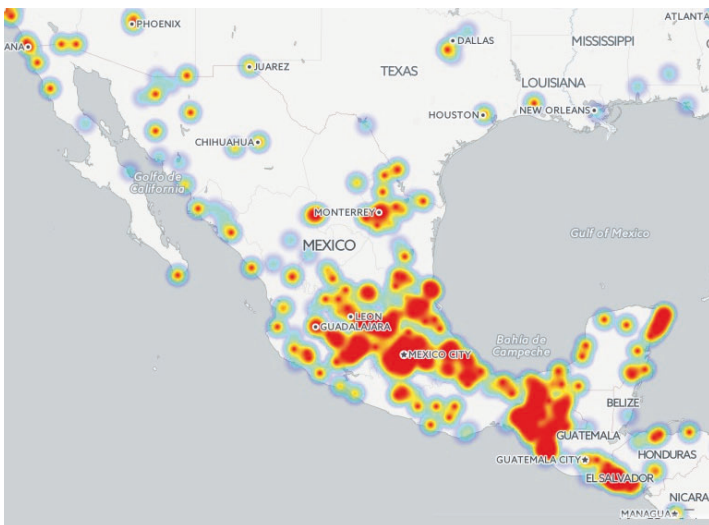


Fig. 3: A geographical heat map showing the distribution of tweets in Mexico. This map clearly highlights the regions of high density and effectively summarizes the important regions in our dataset. In this case, blue color denotes low activity, and red color denotes high activity.

Having described the most active regions in terms of political participation in and having visualized the way in which political participation developed, we turn to analyse the way in which people participate through Twitter. As in traditional expressions of political participation, online social activism can take different forms. In Twitter political participation involves engaging in the online social network through tweets.

Particularly in [8] is pointed out that Twitter users engage with others by addressing users in a conversation through the @ sign and by generating conversational tags through the use of hashtags. In order to identify political participation in Twitter, we begin by building a mentions network. In Fig. 4, we refer to users through nodes. If user i mentions user j in her message, we draw a link between node i and node j . The size of the node represents the number of times the node has been mentioned.

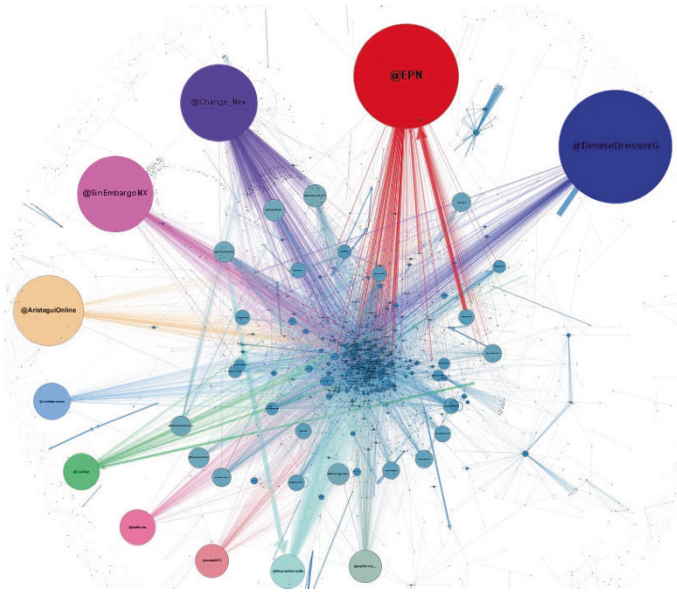


Fig. 4: Network of political participation in Twitter.

Through the visualization of the network it is possible to assess active users in the Twitter mention network. We see that users @EPN, @DeniseDresserG, @Change_Mex, @SinEmbargoMX and @AristeguiOnline are the most mentioned in the sample. This first approximation is useful to categorize the important users within the network. The first category are news agencies, which are represented by @SinEmbargo and @AristeguiOnline. The second category are online social activism groups and social activists, represented by @Change_Mex. Finally, the third category is represented by political pundits like @DeniseDresserG.

Users in Twitter typically organize themselves around specific interests, such as a sports team or hobbies, which facilitates interactions with other users who share similar preferences. These users classify their tweets using topic-specific hashtags [7]. Tweets that contain hashtags entities are inherently more valuable in terms of embedding extra information and bridging knowledge [9]. With this in mind, we identify the online communities that are related to the president of Mexico, Enrique Peña Nieto.

According to our dataset it was possible to identify the most prevalent hashtags appearing in our sample. These topics are as follows: #SinCuotasNiCuates, #Ya-

CholeConTusQuejas and #LeyFayad. The first one is related to an online petition that intends to stop Enrique Peña Nieto from nominating people close to him to the highest court in Mexico: the Suprema Corte de Justicia (Supreme Court of Justice). The second hashtag is related to a TV spot that intended to communicate to the viewers that people is tired of complaints against the government. The third hashtag is related to a law proposal put forward by Omar Fayad. The so called Ley Fayad (Fayads Law) intended to restrict online freedom of speech.

In Fig. 5 we identify large differences on the way in which people participate within different online communities. As can be appreciated within #SinCuotasNiCuates there is a user that is being mentioned by most of the other users. This is Change.org, the online social activism website. In contrast, there are different users that concentrate mentions but in a smaller scale. These users are political pundits such as Julio Astillero (@julioastiller) and Camacho (@CartonCamacho), and social activists such as Enrique D. (@kikesma) and different news agencies just like the magazine Proceso (@revistaproceso).

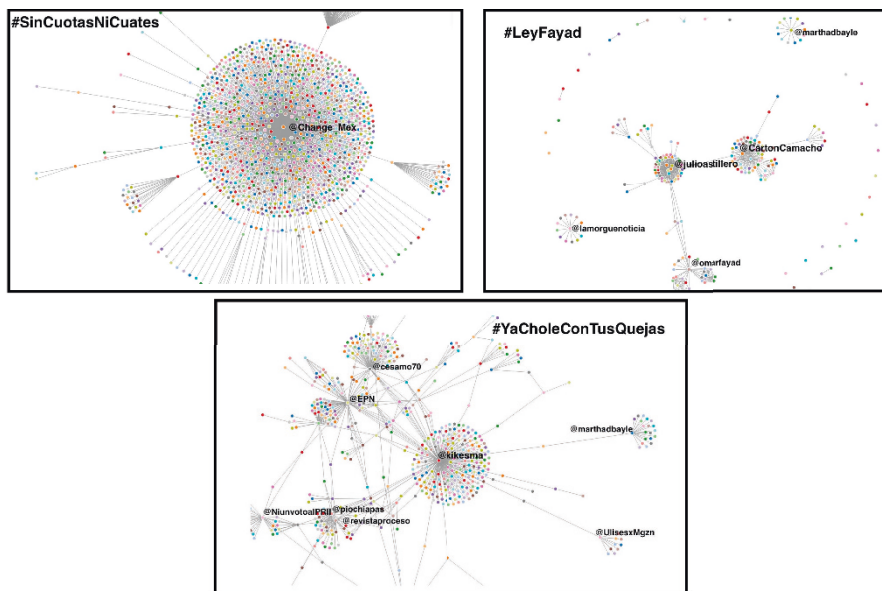


Fig. 5: Mentions network for the online community #SinCuotasNiCuates, #YaCholeConTusQuejas and #LeyFayad.

Our case study of Twitter data suggests that offline political participation in Mexico can be enhanced through online political participation in the OSN Twitter. This is achieved by enabling communities that are usually known in Mexico for its low levels of political participation to participate through different agencies and hashtags. However, it is important to stress that our observations are limited to our sample and, to be generalized, need to be further validated with more data.

4 Conclusion

This paper examines online and offline political participation in Mexico. Through the use of survey data, our article underscores the low levels of interest Mexicans have in politics. This level of interest reflected in the low level of political participation. In particular, we notice that levels of political participation are dependent on the election cycle and, at the same time, regions within the country. Moreover, we note that Mexicans receive political information mainly from television, with other sources of information such as newspapers, radio, the internet and online social networks well behind. In terms of political participation, we see that as the level of personal interaction needed to take part in political action increases, participation seems to decrease.

On the other end, the emergence of new technologies such as Twitter facilitate social interaction to levels never seen before. Therefore, we considered important to examine the way in which political participation in Twitter compared to levels of political participation offline. In our sample of tweets, we found that the general level of online political participation seemed to increase. However, political participation online appears to be different from offline political participation. These differences should be taken with caution because our Twitter sample may not be representative of the Mexican population. In this regard, we observed people participated in three online protests: #SinCuotasNiCuates, #YaCholeConTusQuejas and #LeyFayad. These protests differed in their content, duration and agencies involved. Particularly, we noticed that the duration of the protests may well depend on the agencies involved as #SinCuotasNiCuates was organized around @Change_Mex which most likely organized the debate online. This contrasts with #YaCholeConTusQuejas and #LeyFayad, where the online protests were not organized. This lack of organization might have well contributed to their demise.

Taken together, this study underscores the potential of using social media analysis to develop insight into encouraged users to share political views, and opens the possibility to understand and compare public participation on various scales. Moreover, we show that emergent topics related to politics in Mexico are important because they could help to explore how political participation becomes of public interest.

As was mentioned before, most of our analysis is merely exploratory and, thus, poses questions for future research. Such questions include How do the agencies contribute on the emergence and duration of online protests? and How does online social activism translate into offline activism?

Acknowledgements Carlos Adolfo Piña-García was partially supported by SNI membership 69310.

References

- [1] Almond, G.A., Verba, S.: *The civic culture: Political attitudes and democracy in five nations*. Princeton University Press (2015)
- [2] Golbeck, J.: *Analyzing the social web*. Newnes (2013)

- [3] González-Bailón, S., Borge-Holthoefer, J., Rivero, A., Moreno, Y.: The dynamics of protest recruitment through an online network. *Scientific reports* **1** (2011)
- [4] Kumar, S., Morstatter, F., Liu, H.: *Twitter data analytics*. Springer (2014)
- [5] McIver, D.J., Hawkins, J.B., Chunara, R., Chatterjee, A.K., Bhandari, A., Fitzgerald, T.P., Jain, S.H., Brownstein, J.S.: Characterizing sleep issues using twitter. *Journal of medical Internet research* **17**(6) (2015)
- [6] Norris, P.: La participación ciudadana: México desde una perspectiva comparativa. ponencia magistral presentada el **15** (2002)
- [7] Olson, R.S., Neal, Z.P.: Navigating the massive world of reddit: Using backbone networks to map user interests in social media. *PeerJ Computer Science* **1**, e4 (2015)
- [8] Pearce, W., Holmberg, K., Hellsten, I., Nerlich, B.: Climate change on twitter: Topics, communities and conversations about the 2013 ipcc working group 1 report. *PloS one* **9**(4), e94,785 (2014)
- [9] Russell, M.A.: *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More.* " O'Reilly Media, Inc." (2013)
- [10] Somuano, M., Fernanda, M., et al.: Informe país sobre la calidad de la ciudadanía en México (2014)
- [11] Somuano, M., et al.: *Sociedad civil organizada y democracia en México*. El Colegio de Mexico AC (2007)

Online election campaigning: Identifying political parties using likes and comments

Jessica Liebig, Mohammad Adib Khairuddin and Asha Rao

Abstract Politicians use social media to engage and communicate with voters, in particular during election campaigns. This article investigates data collected from politicians' Facebook pages during the 2013 Australian Federal election and the 2013 Malaysian General election. We wish to gain insight into whether the likes and comments of Facebook users reflect actual connections between politicians during an election campaign. Intuitively, a Facebook user who supports a particular party would not like the posts published by candidates who are associated with opposing parties. However, we observe that users often like the posts by candidates belonging to opposing parties. Our analysis of the data shows that many of the likes and comments made by Facebook users are statistically insignificant. Deletion of these insignificant likes and comments clearly reveals the different parties of the political system. In this paper we consider only the topology of the network representing the datasets, presenting an alternative to the often cumbersome sentiment analysis.

1 Introduction

In today's political landscape, social media plays an important role, in particular during election campaigning [18, 19]. Over the past decade, Facebook, Twitter, YouTube and other social networking sites have become a popular means of communication between political candidates and voters [10]. Politicians commonly use these platforms to make public announcements, in the hope of politically engaging a larger proportion of the population [7].

Past and current research of online election campaigns commonly focus on analysing the language used by politicians [9, 11, 14], the number of likes and comments candidates receive during election periods [3, 9] and the feasibility of us-

Jessica Liebig (e-mail: jessica.liebig@rmit.edu.au)✉ · Mohammad Adib Khairuddin (e-mail: khairuddin.mohammadadib@rmit.edu.au) · Asha Rao (e-mail: asha@rmit.edu.au)

RMIT University, Melbourne, Australia

ing online campaigning to predict the election results [20]. A search of the literature shows that often only basic statistics are used to analyse data from social network sites [6, 12].

In this paper, we take the analysis of social media data one step further by representing it as a complex network. The analysis of online campaigning data as a complex network overcomes some of the limitations of previous studies. For example, comparing the number of likes that different candidates receive during a campaign may lead to false conclusions as not every like is significant. Furthermore, as the authors of [8] point out, a like is not always used as intended and may indeed represent the opposite, that is, disagreement.

By representing two separate data sets, Facebook posts by Australian candidates of the 2013 Australian Federal election and Facebook posts by Malaysian candidates of the 2013 Malaysian General election as bipartite networks, projecting them and extracting the backbone (see Definition 3.1), we demonstrate that we can identify the most significant likes and comments made by Facebook users which then leads to the identification of the different political parties that were well hidden within the network structure.

The rest of the paper is outlined as follows: Section 2 outlines the data collection process and gives a description of the data in the form of some basic statistics. Section 3 provides the necessary background on the identification of significant connections in complex networks. Sections 4 and 5 analyse the Australian Federal election and Malaysian General election, respectively. We conclude the paper by summarising our findings and commenting on future work in Section 6.

2 The data

This paper studies the 2013 Australian Federal election (AFE13) and the 2013 Malaysian General election (MGE13). We extracted posts from the Facebook pages of selected candidates who were part of one of the two elections. The gathered data contains information about the number of likes and comments that posts received as well as the names of candidates who created the posts and the names of users who liked or commented on these. We chose Facebook, rather than other social media platforms, as Facebook allows dialogues between politicians and voters. Voters can send messages directly to politicians and politicians can reach voters through public announcements on their Facebook pages [4, 7].

2.1 Collection

We used NodeXL, an add-in for Microsoft Excel that is freely available at <http://nodexl.codeplex.com/>, to extract data from the candidates' Facebook pages. We restricted the data collection to particular candidates and the respective campaigning periods: 35 days for the Australian Federal election (4th August 2013 - 7th September 2013) and 33 days for the Malaysian General election (3rd April

2013 - 5th May 2013). Only candidates with an active Facebook page were included in this study. The final data sets contain all posts from 55 Australian candidates and all posts from 51 Malaysian candidates during the respective campaigning periods. When the data was initially collected our main interest lay in the 2013 Malaysian General election. As Facebook is not very popular among Malaysian politicians our sample of candidates from Malaysia's general election is relatively small. The Australian Federal Election served as a comparison and hence, the sample of Australian candidates was kept approximately the same size, although Facebook is a popular means of communication amongst Australian politicians. About 45% of the candidates were chosen based on the seats they were contesting. We chose candidates who were contesting marginal seats (seats that are held with less than 56% of the votes) where the outcome of the election is greatly uncertain. The rest of the candidates were selected randomly.

2.2 Basic statistics

Extracting every post that was made by each of the chosen candidates during the campaigning period resulted in two datasets, with 3,608 posts made by the Australian politicians and 8,348 posts made by the Malaysian politicians.

Table 1 gives further information about the collected posts in the form of some basic statistics.

Table 1: Basic statistics of the extracted data.

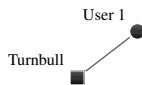
	AFE13	MGE13
Number of posts	3,608	8,348
Average number of posts per candidate	66	164
Total number of likes	371,092	2,512,248
Average number of likes per Facebook page	6,747	49,260
Total number of comments	81,884	387,501
Average number of comments per Facebook page	1,489	7,598

2.3 Network construction

There are several ways of constructing a network from the data. The structure of the data is clearly bipartite, with Facebook users forming the primary node set and political candidates forming the secondary node set. Facebook users actively form



(a)



(b)

Fig. 1: User 1 comments on a post by AFE13 candidate Malcolm Turnbull. Therefore, the user is connected by an edge to Turnbull in the bipartite network representation of the data.

connections to posts and candidates by liking or commenting on posts. One possible representation of the data is as a bipartite network of users and candidates, with an edge existing between a user and a candidate if the user commented on at least one of the candidate’s posts (see Fig. 1). A different representation could have users and candidates connected by likes, rather than comments. It is also possible to construct a network of users and the posts by the candidates, with users being directly linked to posts, rather than candidates. Table 2 lists the different possible bipartite network representations of the data that are considered in this study.

Table 2: A list of the different bipartite network representations that are examined in this study. U denotes the primary node set, V denotes the secondary node set, and E the set of edges. Note that $|E|$, ie. the number of likes/comments, does not match the numbers displayed in Table1, as a user is linked to a candidate if he liked/commented on at least one of the candidate’s posts that is, the first time he likes/comments, and not afterwards.

Name	U	V	E	$ U $	$ V $	$ E $
AFE _{UCL}	Users	Candidates	Likes	119,355	55	143,870
AFE _{UCC}	Users	Candidates	Comments	56,495	55	66,240
MGE _{UCL}	Users	Candidates	Likes	541,726	51	939,395
MGE _{UCC}	Users	Candidates	Comments	198,729	51	272,349

3 Identifying parties and groups of politicians

Election campaigns on Facebook are centred around the individual politicians rather than their associated parties [7]. Our research shows that despite Facebook campaigns being candidate centred it is possible to identify the different parties of the political system within the network representation of the data by considering only the network's topology.

To identify the parties within the network, we use an approach called backbone extraction [13, 16]. The backbone of a network is defined as follows:

Definition 3.1. The backbone of a network $\mathcal{G}(U, E)$, with node set U and edge set E , is defined as the sub-graph $\mathcal{G}'(U, E')$ of \mathcal{G} , such that the edge set E' of the backbone \mathcal{G}' contains only the most significant edges in E .

Determining the most significant connections is not trivial, but requires sophisticated statistical analysis [13, 16]. Recently, the first and third author introduced a very efficient means of extracting the backbone that considerably reduced the computation time of previous methods [13]. They further demonstrated that the backbone of a one-mode projection reveals groups of nodes that are well connected. One-mode projections are simplifications of bipartite networks that only consider one of the two node sets. For instance, when projecting the bipartite network of users and candidates onto the set of candidates, the set of users is dropped and two candidates are connected if at least one user is connected to both candidates in the bipartite network. As noted in both [13] and [16], extracting the backbone of this projection should reveal the significant connections between candidates and this in turn may allow the identification of the different parties contesting the elections using community detection algorithms. We were interested in checking whether the likes and comments of Facebook users reflected actual connections between politicians during an election campaign.

In [13] an edge was defined as significant if its weight was greater than the mean plus three standard deviations of the approximated weight probability distribution. It was shown that the weight probability distribution follows a Poisson binomial distribution and can be approximated by either the Poisson or Normal distributions.

4 The Australian Federal election

To find the significant connections between candidates of the 2013 Australian Federal election we consider the networks of Facebook users and candidates connected by likes (AFE_{UCL}) and comments (AFE_{UCC}).

4.1 Analysis of likes

To examine whether the likes of Facebook users reflect existing connections between candidates of the Australian Federal election, we project the AFE_{UCL} network onto

the set of candidates. Two candidates are now connected if at least one user liked at least one post of each of the two candidates. The edge connecting the two candidates is given a weight that is equal to the number of users who liked posts by both candidates. This weight is then compared to the expected weight and only included in the backbone if it is significantly larger as per [13].

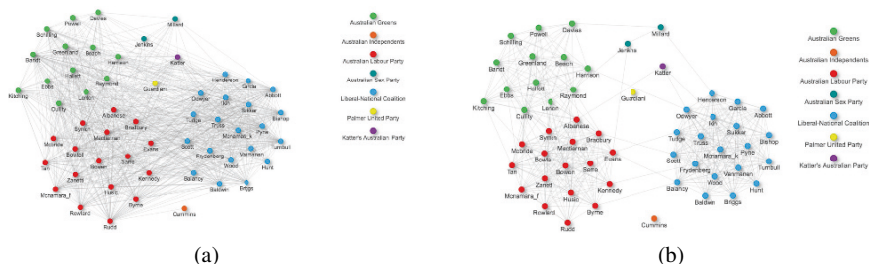


Fig. 2: The projection of the AFE_{UCL} network onto the set of candidates (a) and its backbone (b). The backbone contains 48% of the edges of the projection.

The projection of the AFE_{UCL} network (Fig. 2a) is very dense and a close examination reveals that candidates from different parties are well connected. In other words, users often like posts by candidates from opposing parties. Gerlitz and Helmond [8] state that the like button can express a variety of feelings and may also be used ironically. Hence, simply considering the likes of users does not reveal the candidates who belong to the same party.

4.1.1 Parties

Extraction of the backbone removes most of the connections between candidates of different parties, leading to the conclusion that the connections created by users who like posts by candidates of opposing parties are statistically insignificant. Approximately 52% of edges were identified as insignificant. The backbone of the AFE_{UCL} network (see Figure 2b) clearly reveals the different Australian parties, which were not visible in the projection. Running a community detection algorithm such as the one described in [17] on the backbone produces a list of the different candidates and their associated groups. The community detection algorithm that is introduced in [17] is based on the leading eigenvector of the adjacency matrix and aims to divide the input network into groups such that the modularity is maximised. The modularity of a particular division of a network into groups of nodes can be calculated by subtracting the number of expected edges within these groups if the network was random, from the number of observed edges within the groups. The primary reason for choosing this algorithm is that it has been implemented in the R programming language. Note that the authors of [13] have shown that the backbone of a projection yields higher modularities when running a community detection algorithm than the binary and weighted projections.

The algorithm identifies five groups in total, with all members of the Liberal-National Coalition of Australia being in one group, all Australian Labour candidates being in another and all Australian Greens being in yet another. Guardiani (Palmer United Party) and Cummins (Australian Independents) are isolated nodes and hence, each forms a separate group. Jenkins and Millard (Australian Sex Party) are part of the group that contains the Australian Greens. Katter (Katter's Australian Party) belongs to the same group as the Liberal-National candidates.

The connection between the candidates of the Australian Greens and the Australian Sex Party may be explained by the similarity of the policies of the two parties. For instance, both parties support same sex marriage [1, 2]. The community detection algorithm grouped Bob Katter together with the Liberal-National Coalition. In fact, in 2013 Bob Katter announced his support for a coalition led by Tony Abbott, the former leader of Liberal Party of Australia [5].

4.1.2 Smaller groups of politicians

In [13] an edge was included in the backbone if its weight was greater than the mean plus three standard deviations of the approximated distribution. To identify smaller, well connected groups of politicians within the different parties, here we increase the threshold beyond three standard deviations. Increasing the threshold to five standard deviations separates Katter from the Liberal-National Coalition. An increase to eleven standard deviations results in seven communities, with Millard and Jenkins (Australian Sex Party) forming a separate group. Each of the seven groups now contains politicians from only one party. A further increase of the threshold to 15 standard deviations results in the same seven groups. This shows how well candidates of the same party are connected to each other. Instead of increasing the threshold further to identify groups of politicians within parties, we consider the networks of candidates of each party and the users who like their posts. We extract the backbone with the usual threshold of three standard deviations.

The community detection algorithm identified two groups within each of the Australian Greens, the Australian Labour Party and the Liberal-National Coalition. Interestingly, when extracting the backbone of the projection onto the Liberal-National candidates, all connections between Tony Abbott (Prime Minister, 2013 - 2015) and the other Liberal-National candidates are removed, leaving him isolated and forming one of the groups with all other candidates forming the second group.

Note that we do not know the reason for the divisions within the parties. This would form an interesting research question within the area of political and social sciences.

4.2 Analysis of comments

Projection of the $A_{FE_{UCC}}$ network, where edges represent comments, yields a dense network with candidates of different parties being well connected as was observed in the projection of the $A_{FE_{UCL}}$ network. In contrast to likes however, comments

can usually be categorised into positive and negative. We therefore anticipate that candidates from different parties will also be well connected in the backbone, as a user who supports party A may positively comment on its candidates' posts while negatively commenting on opposing candidates' posts.

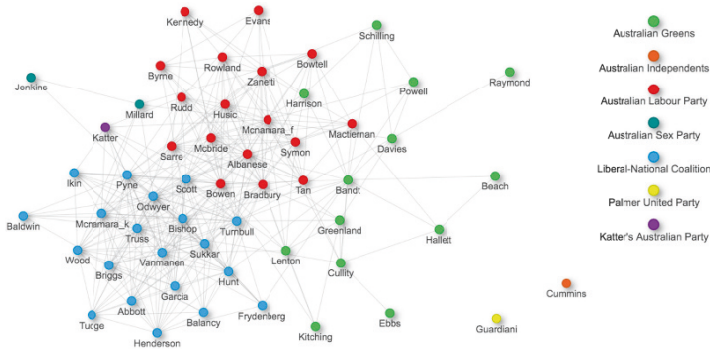


Fig. 3: The backbone of the AFE_{UCC} projection onto the set of candidates. The backbone contains 56% of the edges of the projection.

Running the community detection algorithm on the backbone of the AFE_{UCC} projection reveals the different parties contesting the 2013 Australian Federal election. However, the groups, with many more connections existing between candidates of different parties, are less pronounced than in the backbone of the AFE_{UCL} network (see Fig. 3). Unlike in the backbone of the AFE_{UCL} network, two candidates of the Australian Greens, Harrison and Kitching, are now grouped with candidates of the Liberal-National Coalition and candidates of the Australian Labour Party, respectively. Figure 3 further shows that the Australian Greens are less well connected than in the backbone of the AFE_{UCL} . The greens are amongst those candidates who received the least number of comments, explaining why they are less well connected.

The three most significant connections between members of different parties are the edges between Bradbury (Labour) and Scott (Liberal-National), Albanese (Labour) and Turnbull (Liberal-National), and Bishop (Liberal-National) and Bowen (Labour). Looking at the users who commented on these candidates, we find that they are generally supportive of one of the politicians, responding positively to their posts and leaving negative comments to posts by candidates of the opposing party. We also find that there are users who are against both candidates thus strengthening the connection between opposing candidates. Interestingly, Bradbury and Scott contested the same seat while Turnbull succeeded Albanese as Communications Minister (though in different governments).

5 The Malaysian general election

We repeated our experiments on the Malaysian data. For the 2013 Malaysian General election we considered the networks of candidates and users, connected by likes (MGE_{UCL}) and comments (MGE_{UCC}).

5.1 Analysis of likes

To identify significant connections between Malaysian politicians, we project the MGE_{UCL} network onto the set of candidates. Similar to the projection of the AFE_{UCL} network, the projection of the MGE_{UCL} network is very dense and many connections exist between members of opposing parties.

5.1.1 Parties

Extracting the backbone with a threshold of three standard deviations and running the community detection algorithm reveals two groups of candidates (see Fig. 4). One contains the members of the National Front and Ibrahim Ali (an independent candidate), while the other group contains members of the Democratic Action Party, the Pan-Malaysian Islamic Party and the People’s Justice Party who together form the opposition coalition. From news articles [15], it turns out that the independent candidate Ibrahim Ali was endorsed by the former prime minister and Chairman of the National Front, Dr Mahathir Mohammad, which explains his grouping with the party.

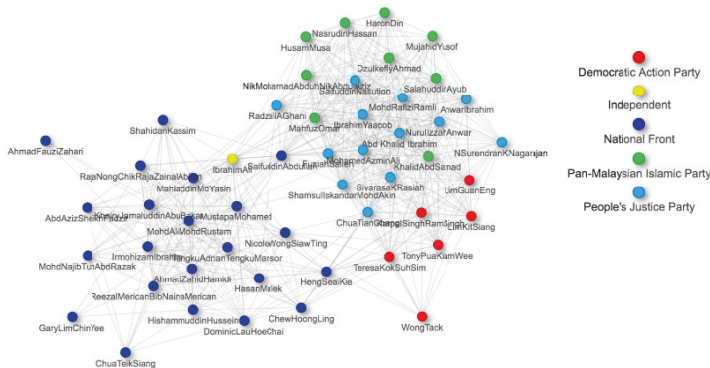


Fig. 4: The backbone of the MGE_{UCL} projection onto the set of candidates. The backbone contains 45% of the edges of the projection.

An interesting node is Saifuddin Abdullah (National Front) who has significant connections to both of the identified groups. As reported in [15], Saifuddin Abdullah is a known progressive-thinking candidate, which does explain his support base extending to both sides of politics. He has also made statements that are in conflict

with the National Front [15]. In addition, two years after the elections, in 2015, he joined the People’s Justice Party, part of the opposition coalition.

5.1.2 Smaller groups of politicians

Extracting the backbone of the network of candidates of each party and the users who like their posts uncovered two groups within the National Front, one of them containing the current Prime Minister, Mohd Najib Tun Abd Razak (assumed office in 2009), and nine other politicians. Eight of the ten candidates in this group contested urban seats, whereas half of the candidates in the second group contested rural seats. No groups could be identified within the other parties.

5.2 Analysis of comments

As was the case with the Australian Federal election, the candidates of the Malaysian General election who belong to different parties are more connected in the backbone of the MGE_{UCC} network (see Fig. 5) than in the backbone of the MGE_{UCL} network.

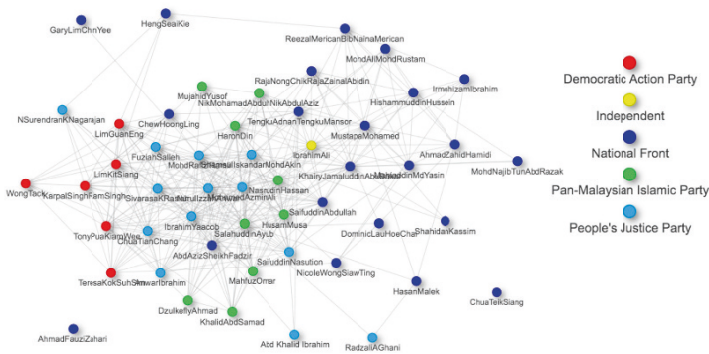


Fig. 5: The backbone of the MGE_{UCC} projection onto the set of candidates. The backbone contains 39% of the edges of the projection.

The community detection algorithm [17] identified five groups of politicians in the backbone of the MGE_{UCC} network. Ahmad Fauzi Zahari and Chua Teik Siang are both isolated nodes and hence each forms a group on his own. The third group consists of politicians associated with the National Front along with a candidate from the People’s Justice Party. The fourth group consists of members of the Pan-Malaysian Islamic Party, three members of the National Front and one member of the People’s Justice Party. The fifth group consists of members of the Democratic Action Party, members of the People’s Justice Party, three candidates of the National Front and three members of the Pan-Malaysian Islamic Party.

The three most significant connections between members of opposing parties are the edges connecting a particular member of the National Front, Abd Aziz Sheikh

Fadzir with Mohd Rafizi Ramli, Mohamed Azmin Ali, and Ibrahim Yaacob, three members of the People's Justice Party. We found that the users who connected Abd Aziz with the other 3 candidates were mostly in favour of one of the candidates. In addition there were users who posted neutral comments to both politicians' posts.

6 Conclusion and future work

Previous research has shown that the 'like' button is not always used as intended and may be used to express a range of feelings. In certain cases it may be used ironically, with a user hitting like in protest [8]. Our analysis of Facebook likes confirmed that users often like political candidates of opposing parties, hence creating connections between them. By using backbone extraction, we were able to show that these connections were mostly statistically insignificant. Retaining only the significant connections clearly revealed the parties that are contesting the election as well as divisions within some of the parties. Identifying the reasons for divisions within parties requires further research, probably something more suited to political scientists.

Users commenting on posts made by candidates of opposing parties was a common observation. Our analysis revealed that these users were generally supporting one candidate and leaving negative comments to the opposing candidate's posts. The work presented in this paper is a first step towards identifying positive and negative posts using complex networks tools, presenting an alternative to sentiment analysis. While the analysis of the content of comments to determine whether someone is supporting or opposing a candidate is cumbersome, the approach presented here is fast and efficient. Further research in this direction of identifying positive and negative posts will be carried out in the future.

References

- [1] Australian Greens, "Our Policies." [Online]. Available: <http://greens.org.au/policy>
- [2] Australian Sex Party, "Our Policies." [Online]. Available: <http://www.sexparty.org.au/policies>
- [3] F. P. Barclay, C. Pichandy, A. Venkat, and S. Sudhakaran, "India 2014: Facebook 'like' as a predictor of election outcomes," *Asian Journal of Political Science*, vol. 23, no. 2, pp. 134–160, 2015.
- [4] J. Bronstein, "Like me! Analyzing the 2012 presidential candidates' Facebook pages," *Online Information Review*, vol. 37, no. 2, pp. 173–192, 2013.
- [5] G. Davies, "Bob Katter and the world." [Online]. Available: <https://independentaustralia.net/politics/politics-display/bob-katter-and-the-world,5826>
- [6] A. Elter, "Interaktion und Dialog? Eine quantitative Inhaltsanalyse der Aktivitäten deutscher Parteien bei Twitter und Facebook während der Landtagswahlkämpfe 2011," *Publizistik*, vol. 58, pp. 201–220, 2013.
- [7] G. S. Enli and E. Skogerbø, "Personalized campaigns in party-centred politics," *Information, Communication & Society*, vol. 16, no. 5, pp. 757–774, 2013.
- [8] C. Gerlitz and A. Helmond, "The like economy: Social buttons and the data-intensive web," *New Media & Society*, vol. 15, no. 8, pp. 1348–1365, 2013.

- [9] R. Gerodimos and J. Justinussen, "Obamas 2012 Facebook campaign: Political communication in the age of the like button," *Journal of Information Technology & Politics*, vol. 12, no. 2, pp. 113–132, 2015.
- [10] T. Graham, M. Broersma, K. Hazelhoff, and G. van 't Haar, "Between broadcasting political messages and interacting with voters," *Information, Communication & Society*, vol. 16, no. 5, pp. 692–716, 2013.
- [11] I. Himelboim, S. McCreery, and M. Smith, "Birds of a feather tweet together: Integrating network and content analyses to examine cross-ideology exposure on Twitter," *Journal of Computer-Mediated Communication*, vol. 18, no. 2, pp. 40–60, 2013.
- [12] A. O. Larsson, "Online, all the time? A quantitative assessment of the permanent campaign on Facebook," *New Media & Society*, vol. 18, no. 2, pp. 274–292, 2014.
- [13] J. Liebig and A. Rao, "Fast extraction of the backbone of projected bipartite networks to aid community detection," *EPL*, vol. 113, no. 2, p. 28003, 2016.
- [14] A. Livne, M. P. Simmons, E. Adar, and L. A. Adamic, "The party is over here: Structure and content in the 2010 election," *ICWSM*, vol. 11, pp. 17–21, 2011.
- [15] Malaysiakini, "Ibrahim Ali out, Saifuddin Abdullah in." [Online]. Available: <https://www.malaysiakini.com/news/226970>
- [16] Z. Neal, "The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors," *Social Networks*, vol. 39, no. 1, pp. 84–97, 2014.
- [17] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, p. 036104, 2006.
- [18] E. Skogerbø and A. H. Krumsvik, "Newspapers, Facebook and Twitter," *Journalism Practice*, vol. 9, no. 3, pp. 350–366, 2015.
- [19] T. L. Towner, "All Political Participation Is Socially Networked?: New Media and the 2012 Election," *Social Science Computer Review*, vol. 31, no. 5, pp. 527–541, 2013.
- [20] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Election forecasts with Twitter: How 140 characters reflect the political landscape," *Social Science Computer Review*, vol. 29, no. 4, pp. 402–418, 2010.

Journalistic Relevance Classification in Social Network Messages: an Exploratory Approach

Miguel Sandim, Paula Fortuna, Alvaro Figueira and Luciana Oliveira

Abstract Social networks are becoming a wide repository of information, some of which may be of interest for general audiences. In this study we investigate which features may be extracted from single posts propagated throughout a social network, and that are indicative of its relevance, from a journalistic perspective. We then test these features with a set of supervised learning algorithms in order to evaluate our hypothesis. The main results indicate that if a text fragment is pointed out as being interesting, meaningful for the majority of people, reliable and with a wide scope, then it is more likely to be considered as relevant. This approach also presents promising results when validated with several well-known learning algorithms.

1 Introduction

Nowadays social networks have become popular systems for sharing and exchanging messages between users. This high rate of information has also turned into a great source of potential, and interesting knowledge, that could be used for the creation of valuable information for a wider audience. In fact, much of the available information scattered among different “discussion groups” in social media, might actually be used in news, or in news creation, since thriving topics on most social networks many times reflect important current events which may be of interest for a more generic audience. On the other hand, we also know that more than usually, information in social media is not relevant outside a short circle of users. Users tend also to post private, personal, or just of a very narrow scope information on their “pages”. In this panorama it is important to have systems capable of aiding in the identification of

Miguel Sandim (e-mail: miguel.sandim@fe.up.pt) · Paula Fortuna (e-mail: paula.fortuna@fe.up.pt) · Alvaro Figueira (e-mail: arf@dcc.fc.up.pt)✉
CRACS / INESC TEC and University of Porto, Rua do Campo Alegre, 1021/1055, 4169-007 Porto, Portugal,

Luciana Oliveira (e-mail: lgo@eu.ipp.pt)
CICE / ISCAP & INESC TEC, Polytechnic of Porto, Rua Jaime Lopes Amorim, Porto, Portugal

what might be interesting information to a wider audience. The goal of the present study is, therefore, to develop a classification model that can automatically identify relevant information in text messages on social networks.

The process of deciding if a particular text has relevant information is neither easy, nor objective, but it is, by far, the most important concern in handling information overload and retrieval [13]: what is relevant for one person, might not be relevant for another; what is not relevant now, might be in a few days or even in a few minutes from now; what is not relevant, can gain relevance just by the inclusion of some context. The combination of possibilities is endless. Moreover, the identification of reasons for personal relevancy diverge from person to person, thus consists on a psychological process by which relevance judgments are made [13] and are computationally difficult to be imitated.

Our approach to the detection of relevance is based on a generalized consensus about which information is relevant to be considered a ‘news’ from a journalist perspective. Although, each journalist may have its own writing style, and personal opinion about any subject, there are a set of guidelines which can help him within this process. Different authors ([1, 5, 6]) suggest some criteria to use: negativity, recency, proximity, consonance, unambiguity, superlativeness, personalization, eliteness, attribution, facticity, continuity, competition, cooption, composition and predictability, to name a few.

Research related to information spread was also found to be either based on the structure of the network it is introduced to or generated on, or on the nature of the content in itself. In fact, while [14] ‘gossip’ analysis is based on the structure of the network, that propels information spreading, [7] argues that virality is strictly connected to the nature of the content, and not to the types of edges linking nodes in specific co-occurrence or social pattern networks.

Moreover, research conducted on text virality [7] indicates that common social network metrics alone (e.g. #likes, #retweets) are not sufficient for assessing such a complex phenomenon and, reinforcing the above mentioned criteria, suggest that several virality components should be considered, such as: appreciation, spreading, simple buzz, white buzz, black buzz, raising discussion and controversiality.

Similarly, our system builds on a set of filters capable of detecting a set of unique characteristics that will enable to create a score for each social media post, allowing to discover “information with potential to be relevant”. Some of these unique characteristics have commonalities to research presented in [7] and in [13], namely: ‘controversiality’ and ‘positiveness’, with the later having the same common ground as ‘white buzz’ and ‘reliability’ (or credibility) and ‘recency’, as mentioned in [13]. Other proposed content features add to research being conducted on the field, such as ‘interest’, ‘meaningfulness’ and ‘scope length’, which are further detailed in section 2.3.

In order to build a classification model it is fundamental to have annotated data with instances to train and test. In a previous study [4] workers from Mechanical Turk classified social network messages as “relevant” or “irrelevant”. The proposed system consisted of a social media crawler and respective classification into “relevant” or “not-relevant” information. However, limitations identified in this preliminary stage

of research led to the development of a more robust and comprehensive methodology. Instead of only asking the workers to answer a binary question about relevance, the workers were asked to give other information that could enlighten the process of journalistic relevance detection, namely by extending the text classification process, in order to include the above mentioned relevance cues. The increase of text classification comprehensives and complexity also allowed us to assure a higher level of trust on the gathered human classification. In the next section we describe this method; we present an analysis based on our results, and draw our conclusions about its efficiency.

The paper is structured as follows: Section 2 defines the methodology that was followed throughout this study; Section 3 presents the results obtained from the exploratory analysis; Section 4 explains the transformation of the users answers on “Crowdflower” into a dataset, as well as the features extracted from each text fragment to potentially explain its relevance. Section 5 describes the experimentation process with several supervised learning algorithms and the results obtained; and finally, Section 6 offers an analysis over the developed work, its viability and envisioned future steps.

2 Related Work

3 Methodology

In order to detect relevance (or irrelevance) in text fragments, a methodology is proposed and described in this section. The phases of this methodology are summarised in fig. 1 and include: data crawling from social networks, data pre-processing, human classification with the use of the “Crowdflower” platform and the development of a classification model.

Each of the illustrated phases in fig. 1 is detailed in the next subsections.

3.1 Crawling from Social Networks

The first phase of this methodology consisted on data crawling from social networks. In this case, the text fragments analysed throughout this paper are posts and comments retrieved from two social networks - Twitter and Facebook - using each the corresponding official API. In order to do so, a Java program was developed to interface with the APIs and with a database built in PostgreSQL.

The data was collected between 1st and 4th April 2016 and included Facebook posts and comments and Twitter tweets. Facebook posts may take the form of status, link, image, video, offer or event. A Facebook post type status (mainly text) may be as long as 63206 characters. Facebook posts may receive comments, likes, shares and reactions (love, haha, wow, sad and angry). Post comments and post shares may also receive likes and replies. A Twitter tweet has a 140 character limit and may

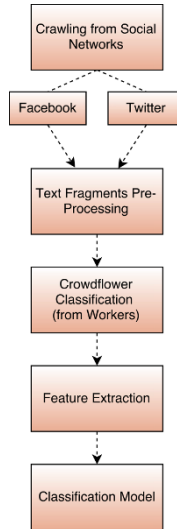


Fig. 1: Pipeline representing the methodology followed.

be marked as favourite and / or be retweeted (which would be the equivalent to a Facebook share).

Data retrieval on twitter was conducted by presenting the API with ten keywords (detailed in section 2.1.1), which were distributed by 100 queries. In what Facebook is concerned, data retrieval was performed on the pages of fourteen international news providers (detailed in section 2.1.2). A maximum of 1000 posts and of 20 comments per post was collected in each news provider page. These difference between the collection methods among the two networks were enforced by restrictions of their own API.

The initial retrieved dataset was composed of 11051 posts, 128673 comments and 76280 tweets.

3.1.1 Twitter

Regarding Twitter, tweets were gathered using the search method provided with one or more keywords from the following list:

- “Refugees” and “Syria”
- “Elections” and “US”
- “Olympic Games”
- “Terrorism”
- “Daesh”
- “Referendum” and “UK” and “EU”

These keywords were chosen based on their popularity in the initial gathering moment, since the probability of fetching a great quantity of tweets in current trending

topics is higher. The search was conducted among the tweets from the previous seven days [3] from the collection moment.

3.1.2 Facebook

In regard to Facebook, the available API did not allow search of posts by keyword. In order to emulate this collection methodology, several posts and comments were collected from fourteen of the most popular international news providers' pages, namely: "Euronews", "CNN", "Washington Post", "Financial Times", "New York Post", "The New York Times", "BBC News", "The Telegraph", "The Guardian", "The Huffington Post", "Der Spiegel International", "Deutsche Welle News", "Pravda" and "Fox News". After the posts and comments collection a search by the keywords was conducted, using the ones specified in section 2.1.1, in order to obtain coherent subject distribution among both networks.

3.2 Text Fragment Pre-Processing

After the crawling from social networks, a control phase was conducted over the gathered text fragments. Since the fragments were extracted for inclusion in a "Crowd-Flower" task, it was important to guarantee that the participants in the task had access to fragments with several quality standards. Therefore, only the text fragments with the following conditions were considered in the sample:

- Number of words between 8 and 100, since if the text fragment is too short in words there may not be enough information to answer the task questions. However if it is too long, it takes too much time and effort for the CrowdFlower's workers to complete the task.
- Written in the English language. A Naive Bayes classifier [10] was used to infer the text fragment's language, assuring homogeneity in the sample.
- With no profanity words, in order to avoid compromising the seriousness of the task.
- Containing all the words from at least one group (from section 3.1.1).
- Not a Twitter "retweet". This assures that all the text fragments are unique.

Other pre-processing actions taken included the removal of links from the text. The complete dataset obtained after the control stage was composed of 1913 comments, 132 posts and 14860 tweets.

The text fragments, as specified in section 3.1.2, include official posts from news channel pages as well as comments in these pages, increasing the probability of having both relevant and irrelevant information in the collected fragments.

Finally, a sample of 101 text fragments was selected in order to assure a higher quality control of the fragments and an equal representativity of each keyword, message type and social network (see Table 2). Some statistics regarding the data selected include: posts from 10 distinct pages, comments from 28 unique users and tweets from 48 unique users. On average posts obtained 3247 likes, 741 shares and

573 comments; an average of 56 likes and 7 replies on comments; and an average of 2 favourites and 4 retweets on tweets. Facebook messages are composed, on average, by 22 words, while Twitter messages include an average of 17 words.

3.3 Crowdfower Classification

In order to perform the relevance classification of the dataset, the selected social network messages were incorporated in a classification task in the online platform “Crowdfower”. This platform was chosen over other ones (e.g. Mechanical Turk) because it offers more control over the quality of the experiment and the users working on it.

The “CrowdFlower” task consisted in a list of eight questions that the users (“workers”) had to answer about the journalistic relevance of a text fragment (see Table 1). The questions were compiled based on the journalistic criteria to find relevant information previously presented ([1, 5, 6]).

Table 1: Questions used in the “Crowdfower” experiment.

Relevance Criteria	Question
“Interesting”	Is the topic of the fragment “not interesting” or “interesting”?
“Controversial”	Is the topic of the fragment “not controversial” or “controversial”?
“Positive”	Is the fragment “negative” or “positive”?
“Meaningful”	Is the fragment “private/personal” or “meaningful for the majority of people”?
“New”	Is the information in the fragment “already known” (for the majority of people) or “new”?
“Reliable”	Is the information in the fragment “unreliable” or “reliable”?
“Wide Scope”	Has the information in the fragment a “narrow” or “wide” scope?
“Relevant”	Is the information in the fragment “irrelevant” or “relevant”?

Each of these questions allowed integer answers in a 5 point Likert scale.

One advantage of the “CrowdFlower” platform, as stated before, is the quality assurance among the “workers” in a task. In this study the following conditions were assured:

- Each fragment was classified by 7 different users, in order to analyze the consensus and subjectivity in the task.
- Each user classified at most 10% of the total fragments, because it was desirable to have as much as variability of participants as possible.
- Only Level 3 “CrowdFlower” users could complete this task. This is the best quality allowed in the platform and relates to the performance of the “workers” on test questions [2].
- All users were either from the UK or the USA, in order to control cultural differences.
- It was assured that each user took at least 20 seconds to complete the job, toward avoiding random and unconsidered answers.

After the experiment in “CrowdFlower” was concluded, a dataset was obtained with the text fragments and its classifications. A sample summary is presented in the next subsection.

3.4 Sample Summary

As a result of the previous phases, a total of 707 answers from 82 different users were collected. Regarding the characterization of this sample, 101 text fragments from 10 news providers’ pages were included and the distribution of text fragments by keyword and message type is detailed in Table 2.

Table 2: Number of text fragments from each group of keywords and social network.

Keyword	FB Posts	FB Comments	TW	Tweets
“Refugees” and “Syria”	5	5	5	8
“Elections” and “US”	5	5	5	8
“Olympic Games”	2	5	5	8
“Terrorism”	5	5	5	8
“Daesh”	2	5	5	8
“Referendum” and “UK”	4	5	5	8
“EU”				

4 Exploratory Analysis

In order to better understand the process of relevance classification, an exploratory analysis was conducted using Pearson Correlation. The results of this analysis are presented in Table 3.

Table 3: Correlations between all the questions and the “Relevant” question for the 707 answers.

	“Relevant”	
“Interesting”	<i>r</i>	0.61
	<i>p</i>	<0.001
“Controversial”	<i>r</i>	0.24
	<i>p</i>	<0.001
“Positive”	<i>r</i>	0.12
	<i>p</i>	<0.001
“Meaningful to the Majority”	<i>r</i>	0.60
	<i>p</i>	<0.001
“New”	<i>r</i>	0.15
	<i>p</i>	<0.001
“Reliable”	<i>r</i>	0.60
	<i>p</i>	<0.001
“Wide scope”	<i>r</i>	0.65
	<i>p</i>	<0.001

The correlations and *p* values indicate that the more the information is “interesting”, “meaningful for the majority”, “reliable” and with a “wide scope”, the more it is perceived as being “relevant” by the evaluators.

5 Surrogate Feature Extraction

In the previous section some characteristics of the information were presented as indicators of relevance in text fragments. However these variables were dependent on human classification and in order to classify a text fragment as “relevant” or “irrelevant” these features must be extracted automatically from the text or social network information. Therefore, several features were added aiming at replacing each question.

Table 4: Conversion between questions and automatic features.

Relevance Criteria	Goal	Surrogate Features	Description
Interesting	This group of metrics is based on the idea that people will react and share more information if it is interesting.	Number of user mentions	Number of “@” used in the text fragment to refer other users in the same social network
		Number of likes	Number of favorites in a tweet or number of likes in posts or comments from Facebook
		Number of shares	Number of “retweets” of a tweet or the number of shares of a Facebook post
		Comment count	Number of comments of a Facebook post and is not applicable to Twitter
Personal vs. Meaningful	Evaluate the subjectivity in the text fragment.	Sentiment Analysis [9]	Processed with the “polarity” function from the package QDAP [11] in R
		Number of Adjectives	Indicator for higher subjectivity [12]
		Number of pronouns (in first or second person)	Referred as an indicator for relevance [4]
Reliability	Use the credibility of the owner of the message.	Verification status	Status (verified or not) of the Facebook/Twitter profile that published the text fragment
		Number of followers	Number of followers of the Twitter profile or number of likes in a page from Facebook

5.1 Relation between relevance criteria and surrogate features

Aiming at evaluating the potential of automatic classification of relevance, a set of surrogate features matching the pre-established relevance criteria were extracted and developed, as represented in Table 4. In order to do so, social media metrics and additional methodologies were incorporated. At this stage, it was possible to correlate three of the relevance criteria with several automated processes. For instance, a set of surrogate social media metrics, such as number of user mentions, number of likes, shares and comments, can be indicative of ‘interesting’ content. Likely, performing sentiment analysis as well as adjective and pronoun counting can assist on evaluating the subjectivity of the messages. Finally, the verification status and the number of followers can be surrogate features for the relevance criteria ‘reliability’.

5.2 Journalistic Relevance Class

Regarding the “Relevance” question, the numeric answer was converted into categorical. Each answer was transformed into a class according to the following rule: 1 or 2 became “Irrelevant”, 3 became “Neutral” and 4 or 5 became “Relevant”. Since each text fragment was classified by 7 users several agreement ratios were analysed (see fig. 3).

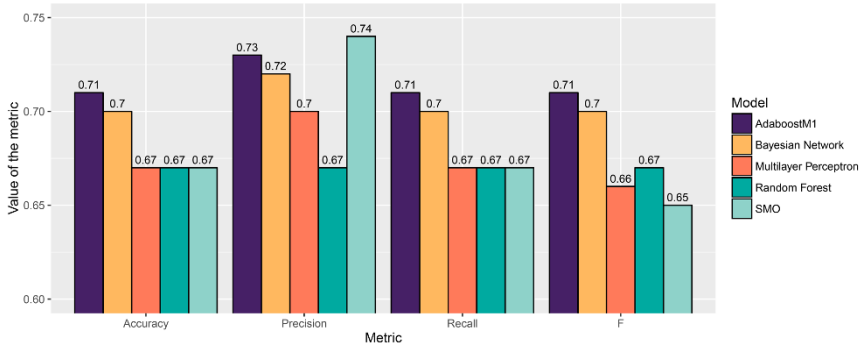


Fig. 2: Accuracy, precision, recall and F measure for each supervised learning algorithm.

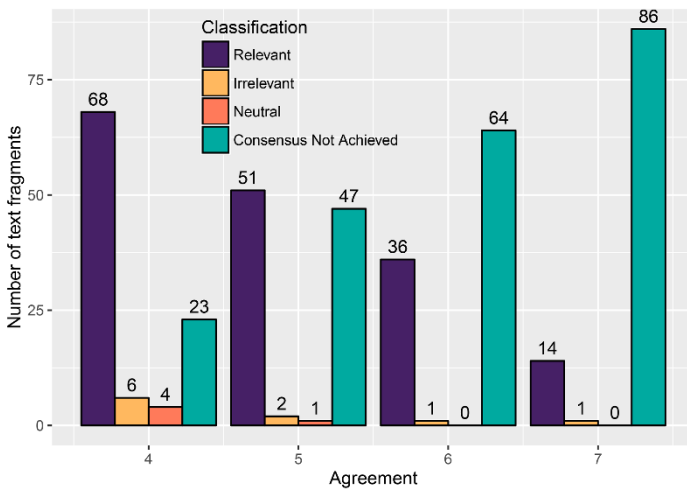


Fig. 3: Number of text fragments in each categorical answer (“Irrelevant”, “Neutral” or “Relevant”) with different agreement ratios.

In order to balance the number of instances in each class, the chosen agreement value was 5: a text fragment was considered “Relevant” if at least 5 workers answered “4” or “5” for the text relevance question. In any other case (“Irrelevant”, “Neutral” or “Consensus Not Achieved”) the text fragment was considered “Not Relevant”. Therefore with this criteria the number of text fragments considered as “Relevant” and “Not Relevant” was 51 and 50 respectively.

6 Classification Model

In order to understand the importance of each feature, the “Relief F” metric [8] was computed. The results revealed that the message type (which distinguishes “FB Posts” from “FB Comments” and “Tweets”), the number of comments (if applicable) and the verified status of the author of the text fragment are the most influential attributes. The feature ranking obtained with this metric is presented in Table 5.

Table 5: Relief F attributes with value greater than “0”.

Features	Ranking Value
message type	0.15
comment count	0.13
verified	0.06
followers count	0.01
shares	0.01

Several experiments with different models were also conducted, with “AdaboostM1” and “Bayesian Networks” being the algorithms which achieved higher accuracy (71% v.s. 70%) and F score (71% v.s. 70%). These results are summarised in fig. 2.

7 Conclusion

In this paper we presented an exploratory study about relevance classification in a journalistic perspective. The first stage of our methodology consisted of: (1) collecting posts from social networks (either from Facebook and Twitter) according to a set of popular, yet controversial, topics; (2) filtering the retrieved posts to gather a dataset with enhanced quality (e.g. with a reasonable quantity of words, written in English, etc); (3) submitting this final set for a classification job in “CrowdFlower”.

Our analysis of the results pointed out that interesting, meaningful, reliable and wide scope information is more likely to be considered as relevant for a majority of 5/7 of workers. This exploratory analysis led us to identify surrogate features, which could be accessed/extracted, or computed, automatically to predict relevance.

In a second stage we applied five machine learning algorithms to our golden standard. In almost all metrics (accuracy, precision, recall and F-value) the “Bayesian Networks” and the “AdaboostM1” have the best performance for the available data. Regarding the features used, we found out that “message type” and “comment count” are the most important ones for this analysis. Besides, the significant correlations, the accuracy and the F-value showed that the quality control validated the proposed methodology to detect relevance in social network messages.

Finally, for the future work two different goals could be considered. Firstly it is important to increase the sample size of classified messages with the intent of

strengthening the confidence in the methods used. Secondly new surrogate features should be researched (e.g. related with the wide scope of the information in the text fragments) to complete the automatic classification relevance model.

Acknowledgements This work is financed by the ERDF European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT Fundao para a Ciênciã e a Tecnologia (Portuguese Foundation for Science and Technology) within project “Reminds/ UTAP-ICDT/EEI-CTP/0022/2014”.

References

- [1] Bell, A.: The language of news media. Blackwell Oxford (1991)
- [2] Crowdflower: Crowdflower community - introducing contributor performance levels! (2014). URL <http://crowdflowercommunity.tumblr.com/post/80598014542/introducing-contributor-performance-levels>. Accessed: 2016-04-28
- [3] Developers, T.: The search api. <https://dev.twitter.com/rest/public/search>. Accessed: 2016-04-21
- [4] Figueira, A., Sandim, M., Fortuna, P.: An approach to relevancy detection: Contributions to the automatic detection of relevance in social networks. In: *New Advances in Information Systems and Technologies*, pp. 89–99. Springer (2016)
- [5] Galtung, J., Ruge, M.H.: The structure of foreign news the presentation of the congo, cuba and cyprus crises in four norwegian newspapers. *Journal of peace research* **2**(1), 64–90 (1965)
- [6] Gans, H.J.: *Deciding what’s news: A study of CBS evening news, NBC nightly news, Newsweek, and Time*. Northwestern University Press (1979)
- [7] Guerini, M., Strapparava, C., Özbal, G.: Exploring text virality in social networks. In: *ICWSM* (2011)
- [8] Kira, K., Rendell, L.A.: The feature selection problem: Traditional methods and a new algorithm. In: *AAAI*, vol. 2, pp. 129–134 (1992)
- [9] Liu, B.: Sentiment analysis and subjectivity. *Handbook of natural language processing* **2**, 627–666 (2010)
- [10] Nakatani, S.: Language detection library for java (2010). URL <https://github.com/shuyo/language-detection>. Accessed: 2016-04-21
- [11] Rinker, T.W.: qdap: Quantitative Discourse Analysis Package. University at Buffalo/SUNY, Buffalo, New York (2013). URL <http://github.com/trinker/qdap>. 2.2.4
- [12] Rittman, R., Wacholder, N., Kantor, P., Ng, K.B., Strzalkowski, T., Sun, Y.: Adjectives as indicators of subjectivity in documents. *Proceedings of the American Society for Information Science and Technology* **41**(1), 349–359 (2004)
- [13] Sundar, S.S., Knobloch-Westerwick, S., Hastall, M.R.: News cues: Information scent and cognitive heuristics. *Journal of the American Society for Information Science and Technology* **58**(3), 366–378 (2007)
- [14] Tasgin, M., Bingol, H.O.: Gossip on weighted networks. *Advances in Complex Systems* **15**(supp01), 1250,061 (2012)

Part IX
Networks in Finance and Economics

Stock prices prediction via tensor decomposition and links forecast

Alessandro Spelta

Abstract The recent financial crisis has stressed the need to understand financial systems as networks of stocks, where financial linkages can be represented by Euclidean distances between stocks pair. It has also been emphasized that the relevance of these networks relies on the representation of changes follow on the occurrence of stress events. In finance, for instance, market crashes are the consequence of herding behaviors that increase the correlation between the units of the system lowering the distances between nodes and therefore the network links. Consequently, predicting future links between stocks can be a valuable starting point for inferring markets down-turn. This is the scope of the work. It introduces a multi-way procedure to forecast stock prices by decomposing a distance tensor. This multidimensional method avoids aggregation processes that could translate into losses of crucial features of the system. The technique is applied to a basket of stocks composing the S&P500 composite index and to the index itself for demonstrating its ability in predicting large market shifts that arise in the face of turbulences, such as ongoing financial crisis.

1 Introduction

The 2008 financial crisis has shown that network theories can enrich the understanding of financial systems and the comprehension of factors causing failures in financial markets. As a consequence, a growing interest in applying methods from complex networks in financial research has been recently developed. All these methods (see [19], for instance) represent stock markets as correlation based networks where the stocks are the nodes and financial linkages can be represented by Euclidean distances between stocks pair. Furthermore, it has become clear that the relevance of these networks relies on the representation of changes follow on the occurrence of stress events.

Alessandro Spelta (e-mail: alessandro.spelta@unicatt.it)✉
Catholic University and Complexity Lab in Economics, Milan

Financial markets experience sudden regime shifts where fluctuations lead to an increase of the correlation between the units of the system, lowering in this way the distances between the stocks and therefore the network links by creating upward and downward trends. Those changes usually take place at critical thresholds - the so-called tipping points - and are associated with critical transitions between alternative states of the system. Predicting these changes is a difficult task but fortunately some theoretical researches [6, 18] suggest the existence of generic indicators for critical transitions even when the knowledge of the functioning of the systems is insufficient to build up predictive models. The underlying principle of most of these indicators is a phenomenon known in dynamical systems theory as critical slowing down. Beside the growing autocorrelations of the state variables of the system, recent works [5, 10] have suggested that the critical slowing down phenomenon might, in theory, generate also spatial signals such as an increasing spatial correlation near transitions. Such occurrence is due to the fact that the entities composing the system pass from isolated to coordinated behaviors, where a spontaneous order emerges [5, 13]. When the intrinsic dynamics of each entity is weakened, the units will be strongly dependent on that of its neighbors. As a result, units will become more strongly correlated close to the transition. In finance, for instance, the formation and collapse of speculative bubbles have been largely considered as the consequence of herding behaviors emerging from the broken balance between autonomous conducts and peer influence [17]. When the effect of exchanging influence with the rest of the environment dominates, large-scale phenomena occur. Indeed, while during expansion and normal periods financial markets tend toward randomness, in crisis phases their structures are reinforced due to a generalized increase in the level of correlations that leads to a contraction of the linkages between stocks in the correlation based networks [1, 14]. Although there exists empirical evidences of connections between strengthening of links in the stocks networks and crisis episodes in financial markets, most of the existing studies mainly focus on correlations between stock prices [15], the resulting distance based networks and on their Minimum Spanning Trees representations [19], to provide optimal asset allocations and portfolio risk estimations.

This paper, in turn, explicitly addresses the question of inferring the forthcoming dynamic of stock prices through the forecast of future distances between stocks in correlation based networks. This issue technically amounts to a link prediction problem [12]. Given past links (distances) between stocks, what will be their next period value? If predictions suggest a contraction of the next period distances for instance, then we could expect a decrease in stock prices because of a strengthening in correlations and a higher likelihood of a crisis episode.

The mainstreaming class of link prediction methods, are based on the so-called similarity-based algorithms, which are further classified into three categories: local, global and quasi-local depending on the information used [12]. Usually all these techniques collapse the temporal data into a single matrix by summing (with or without weights) the records corresponding to the temporal networks. Then similarity-based measures like the Katz centrality or the singular value decompositions (SVD) are applied to perform links prediction. This paper instead is the first attempt to use

tensor decompositions and multi-way analysis [7, 9] to extract complex relationships from stock prices' time series and use them in a link prediction application. This approach prevents the temporal aggregation of the data, avoiding losses of crucial features of the system that can be observed only by holding the original time-varying nature of the records.

Starting from N time series of stock prices, a rolling window of length n_1 is applied to compute the correlation $C_{k,l}$ among each pair (k, l) of stocks. Given these pairwise correlations, at each time step, a distance based network with elements $d_{kl} = \sqrt{2(1 - C_{k,l})}$ is created.

Once the rolling window has produced Z distance based networks with adjacency matrices $\mathbf{D} \in \mathbb{R}^{N \times N}$, those matrices are embedded into a 3D-tensor $\mathcal{D} \in \mathbb{R}^{N \times N \times Z}$ whose generic element δ_{klz} represents the distance between stock k and stock l at time z .

The tensor is thus approximated as the outer product of three vectors through the Canonical Decomposition [4], also known as Parallel Factorization [8], the so-called CP decomposition, which can be regarded as a generalization of SVD to tensors (see 2).

The decomposition aims at writing the tensor \mathcal{D} as the outer product of two identical vectors \mathbf{v} , that contains the *overall spatial dissimilarity* between stocks and a vector \mathbf{u} , containing the *temporal profile* of the dissimilarities $\mathcal{D} \cong \lambda \mathbf{v} \circ \mathbf{v} \circ \mathbf{u}$ where $\mathbf{v} \in \mathbb{R}^N$, $\mathbf{u} \in \mathbb{R}^Z$ and $\lambda = \|\mathbf{v}\| \|\mathbf{v}\| \|\mathbf{u}\|$.

While a stock with a high (low) overall spatial dissimilarity score has, on average, a different (similar) behavior compared with the one of the rest of the stocks, a period in which a high (low) temporal profile score is registered will be a period in which most of the stock are highly dissimilar (similar).

The next step consists in generating the adjacency matrix of the forecasted distance based network. Instead of predicting the N^2 possible distances using N^2 data points, within this method one has to predict only the next value of the temporal profile \mathbf{u} and use it, together with the two fixed overall spatial dissimilarity vectors \mathbf{v} , to build up the adjacency matrix of the forecasted distance based network. An exponential smoothing, applied to the last n_2 observations of the temporal profile vector \mathbf{u} , extracts a scalar τ representing the guess of the next period value of such vector. Then the adjacency matrix containing the forecasted distances of all stocks pair is obtained as a linear combination of the two spatial dissimilarity vectors \mathbf{v} , the parameter λ and of the forecast τ of the temporal profile vector. In matrix terms; $\widehat{\mathbf{D}} = \tau \lambda \mathbf{v} \mathbf{v}^T$ or, element-wise $\widehat{d}_{kl} = \tau \lambda v_k v_l$ (where the superscript $\hat{\circ}$ denotes the predicted distance). Finally, the vector of the forecasted prices is found as the outer product of current price vector and of the normalized matrix representing the predicted future distances $\widetilde{\mathbf{D}}$. The normalization is obtained by dividing each entry of the predicted distance matrix by the number of the stocks in the dataset. In this way the forecast of a stock price will be equal to the current price multiplied by the average of the predicted distances that relate it to the rest of the stocks. In matrix terms; $\widehat{\mathbf{P}} = \widetilde{\mathbf{P}} \mathbf{D}$ or, element-wise, $\widehat{P}_i = P_i \frac{1}{N} \sum \widehat{d}_{i,:} = P_i \sum \widetilde{d}_{i,:}$.

In accordance with the empirical evidence suggesting links contract during crisis period in distance based networks, the predicted price for each stock will be lower

than the current one if, on average, the distance between that stock and the rest is decreasing.

When the steps of the moving window exceed the parameter Z , the tensor is allowed to move in time at each new step, as new data are available. The temporal shift of the tensor permits to compare the forecasts produced by two consecutive decompositions¹. The difference between the values of the two predictions generates a signal whose sign indicates the future direction of the price.

To investigate whether this method is able to correctly identify changes in stock prices a backtest based on a hypothetical investment strategy is implemented [16]. If the sign of the signal for a given stock i is negative, a short position is taken by selling the stock and buying back it the next trading day. In this case, the cumulative return made on that stock R_i changes by $\frac{P_i^t - P_i^{t+1}}{P_i^{t+1}}$. Otherwise, if the difference is positive, a long position is taken by buying the stock and then selling it the next trading day. The cumulative return in this case changes by $\frac{P_i^{t+1} - P_i^t}{P_i^t}$. Notice that profits are only possible if at least some future changes in stocks prices are correctly anticipated, in particular around large market movements. Fig. 1 gives a graphical representation of the technique.

2 Materials and methods

Tensor decompositions and multi-way analysis can be naturally employed to represent the time-varying distance matrices as a single mathematical object, a three-way tensor, and approximate this tensor as a product of vectors by extracting the most relevant spatial and temporal factors [4, 8]. Uncovering the spatial (\mathbf{v}) and the temporal profile (\mathbf{u}) vectors that contains the overall dissimilarities between stocks and the related activity pattern requires the identification and the extraction of lower-dimensional features. This can be achieved by means of the so-called canonical CP decomposition in three dimensions.

The decomposition aims at writing the tensor \mathcal{D} as the outer product of two identical vectors \mathbf{v} , that contains the *overall spatial dissimilarities* between stocks' time series and a vector \mathbf{u} , containing the *temporal profile* of the dissimilarities: $\mathcal{D} \cong \lambda \mathbf{v} \circ \mathbf{v} \circ \mathbf{u}$ where $\mathbf{v} \in \mathbb{R}^N$, $\mathbf{u} \in \mathbb{R}^Z$ and $\lambda = \|\mathbf{v}\| \|\mathbf{v}\| \|\mathbf{u}\|$.

Such an approximation of the tensor \mathcal{D} is equivalent to minimizing the Frobenius norm of the difference between \mathcal{D} and $\lambda \mathbf{v} \circ \mathbf{v} \circ \mathbf{u}$. Solving this problem amounts at finding the rank-1 tensors that best approximate the \mathcal{D}

$$\min_{\mathbf{v}, \mathbf{u}} \|\mathcal{D} - \lambda \mathbf{v} \circ \mathbf{v} \circ \mathbf{u}\| \quad (1)$$

The 3-dimensional problem is divided into 3 sub-problems by unfolding the tensor \mathcal{D} . This means reordering the elements of a tensor into a matrix. The mode-3 unfolding

¹ The price forecasts are not compared with their current values due to a bias given by the fact that, when the correlation is zero, the distance takes a value of $\sqrt{2}$ artificially incrementing the price of the stocks.

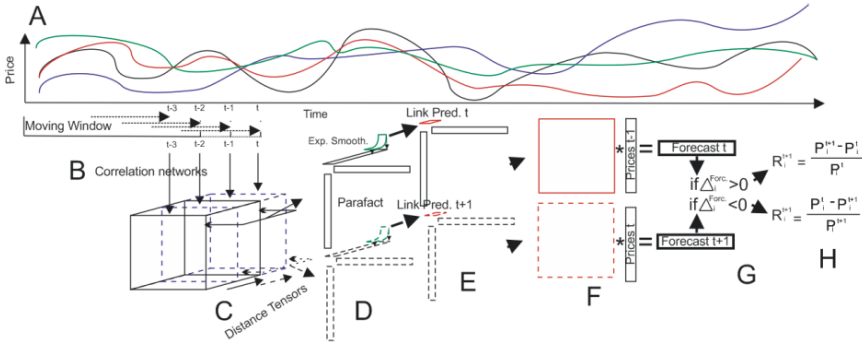


Fig. 1: Graphical representation of the method. Starting from stock price time series (A), a rolling window is applied to compute the correlation among each pair of stocks (B). At each time step a distance based network is created. Once the rolling window has produced Z distance matrices representing adjacency between stocks pairs, those matrices are embedded into a 3D-tensor \mathcal{D} (C). When the steps of the moving window exceed Z , the tensor is allowed to move in time at each new step, as new data are available (C - solid line vs. dashed line). The two consecutive tensors are approximated as the linear combination of three vectors $\mathcal{D} \cong \lambda \mathbf{v} \circ \mathbf{v} \circ \mathbf{u}$. The exponential smoothing (D - green lines) applied to \mathbf{u} extracts a scalar τ representing the forecast of temporal profile for the next period (E - red lines). The forecast of the future distance matrix is obtained as $\widehat{\mathcal{D}} = \tau \lambda \mathbf{v} \mathbf{v}^T$ (E and F - red squares). Finally, the prediction of future prices is computed as the outer product of the past price vector and of the normalized matrix representing the predicted distances $\widehat{P}_i = P_i \frac{1}{N} \sum \widehat{d}_{i,:}$. (F and G). An investment strategy is proposed to assess the efficiency of the method. If the difference $\Delta_i^{Forc} = \widehat{P}_i^t - \widehat{P}_i^{t-1}$ between two consecutive price forecast for a generic stock i is negative (G) then a short position is taken. Otherwise a long position is taken. The returns are calculated as $\frac{P_i^t - P_i^{t+1}}{P_i^{t+1}}$ or as $\frac{P_i^{t+1} - P_i^t}{P_i^t}$ depending whether a short or a long position is taken (H).

of a tensor \mathcal{D} is denoted by $\mathbf{D}_{(q)}$ and arranges the mode- q fibers to be the columns of the resulting matrix. For the 3-D case, the three resulting matrices have respectively a size of $N \times NZ$, $N \times NZ$ and $Z \times N^2$. In this way problem 1 is equivalent to minimizing the difference between each of the modes and their respective approximation in terms of factors. Problem 1 is thus converted into three problems

$$\begin{aligned}
 & \min_{\mathbf{v} > 0} \left\| \mathbf{D}_{(1)} - \lambda \mathbf{v} (\mathbf{u} \odot \mathbf{v})^T \right\|_F^2 \\
 & \min_{\mathbf{v} > 0} \left\| \mathbf{D}_{(2)} - \lambda \mathbf{v} (\mathbf{u} \odot \mathbf{v})^T \right\|_F^2 \\
 & \min_{\mathbf{u} > 0} \left\| \mathbf{D}_{(3)} - \lambda \mathbf{v} (\mathbf{v} \odot \mathbf{v})^T \right\|_F^2
 \end{aligned} \tag{2}$$

where \odot denotes the Khatri-Rao product, namely the column-wise Kronecker product. Since distances are always non negative, a non-negative tensor factorization method is employed to solve (2) because it greatly simplifies the interpretation of the resulting decomposition. The Block Coordinate Descent Method for Regularized Multiconvex Optimization [20] and the Matlab Tensor Toolbox [3] are used to solve (2).

Similarly to the TOPHITS algorithm [11], the overall spatial dissimilarity score of a generic stock i is found as a function of the scores of the rest of the stocks weighted by the product of the distances connecting them to stock i , and of the temporal profile score of the period in which the distances are observed. The temporal profile score attached to a period, on the other hand, is a weighted sum of the distances recorded in that period. Where each distance is weighted by the product of the spatial dissimilarity score of the stocks connected by such distance. In this way, the spatial dissimilarity vectors retain also elements representing the temporal evolution of the distances and only the "next step" value of the temporal profile vector has to be inferred from past data. This is a perspective not available when computing link predictions using matrix-based approaches. A temporal link prediction, naturally follows from the decomposition and can be used to infer future distances between stocks, and, on the basis of these forecasts, to predict future prices.

3 Results

The method is applied to the closure price of a basket of 388 stocks composing the S&P500 composite index, traded during 3527 working days, from 1999/08/04 to 2013/08/09. Additionally, a modified version of the method is also employed to forecast the dynamic of the S&P500 composite index for the period ranging from 2004/06/24 to 2013/04/30. Fig. 2 shows the cumulative sum of the returns obtained for each stock together with the average cumulative performance, namely, the mean of the stocks returns (solid black line). Beside the fact that the investment strategy does not produce positive returns for all the stocks, the values of y-axis, biased in favor of positive quantities, together with the positive average return (black line), that reaches the value of 230% at the end of the sample, confirm the ability of the methodology to produce good predictions. Fig. 2(a) displays the cumulative returns obtained by investing only taking short while the cumulative returns obtained by only taking long positions are showed in Fig. 2(c). In this way one is able to compare the performance of the methodology in predicting down-turns or up-turns of stock prices. From the average performance (black line) reported in Fig. 2(b) clearly emerges that deep crashes, the burst of the dot-com bubble and the 2008 financial crisis, are correctly anticipated. This more than compensate the losses of taking long positions (Fig. 2(c)) during these phases. These simulations are performed using the following parameters: $n_1 = 15$, $Z = 25$, $n_2 = 7$ and the exponential smoothing parameter is set to be equal to 0.2.

While Fig. 2 aims at discording whether the movements of each stock are correctly anticipated by only looking at the sign of the signal produced by the methodology, the next step consists in assessing the quality of the signals. In theory, the larger the

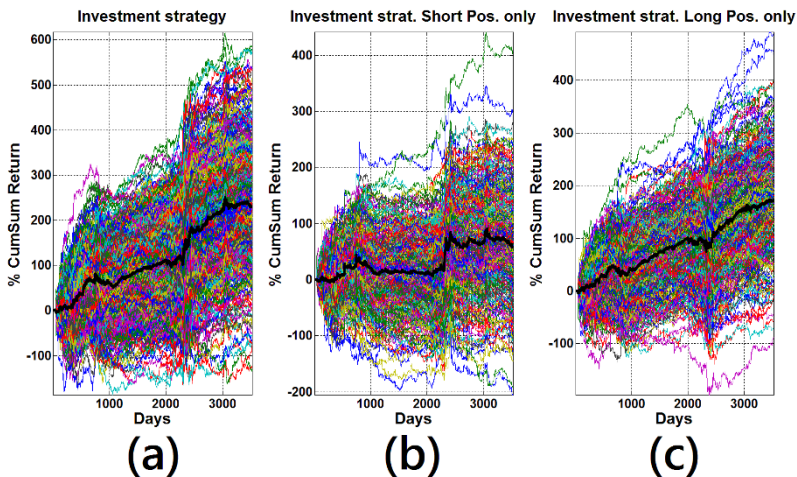


Fig. 2: Cumulative sum of returns obtained for each stock together with the average cumulative performance (solid black line). Fig. 2(a) shows the performance of the investment strategy. Panels (b)-(c) display the cumulative returns obtained by investing only taking short or long positions respectively. This helps in quantifying how the correct forecast of booms and burst phases affects the overall cumulative returns. The simulations are obtained using the following parameters: $n_1 = 15$, $Z = 25$, $n_2 = 7$ and the exponential smoothing parameter is set to be equal to 0.2.

absolute difference between two consecutive forecasts, i.e. the larger the absolute value of a signal, the more credible the forecast is. In order to show this feature, at each time step, the signals are sorted in descending order, based on their absolute values. Fig. 3(a) displays the average cumulative sum of the returns associated with different quantiles of the signals distribution. In particular the upper blue line is associated with the strongest signal, the green line shows the average cumulative returns produced by the two strongest signals, the red line indicates the average performance of the first forty-five signals. The other lines illustrate the performance associated with the cumulative sum of signals of gradually lower quantiles. Finally the lowest purple line displays the average cumulative return for the whole signals distribution (and it is equivalent to the black line of Fig. 2). Also in this case the methodology is able to correctly predict the largest market movements, especially near deep burst phases as shown in Fig. 3(b). Moreover Fig. 3 points out that stronger signals produce better forecast, proving that the cumulative returns associate the most robust signal (436%) doubles the average performance associate to all the signals (230%).

To further analyze the goodness of the proposed methodology, the method is also applied to forecast the behavior of the S&P500 index as a whole. Consequently technique has been slightly modified to produce predictions for the whole composite index and not for each stock constituting the basket. First, the number of stocks in the dataset is augmented (455 stocks are employed in this exercise), by restricting

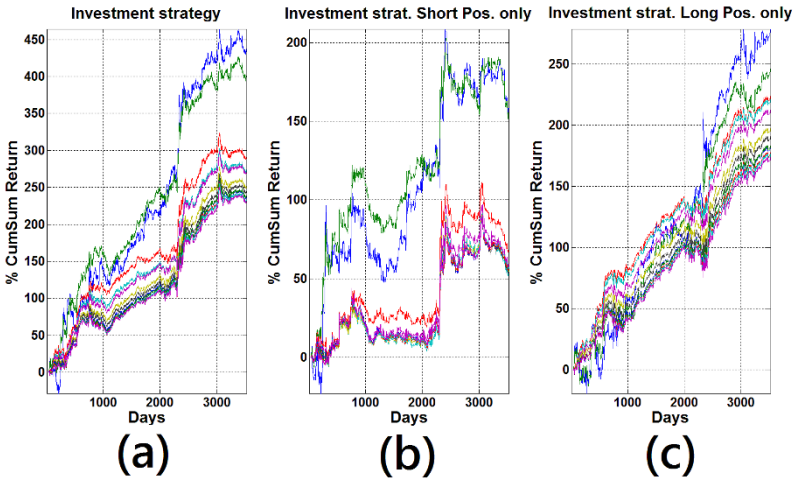


Fig. 3: Cumulative sum of the returns associated with different quantiles of the signals distribution. The goodness of each signal depends on the absolute difference between the two consecutive forecasts that compose the signal. The larger the difference the better the signal. While Fig. 3(a) shows the performance of the whole investment strategy. Panels (b)-(c) display the cumulative returns obtained by only taking short or long positions respectively. The signals are sorted according to their absolute values and therefore there is no a one-to-one correspondence between each plotted line and a particular stock. For instance, the best signal can regard different stocks in different moments in time. The simulations are obtained using the same parameters of Fig. 2.

the temporal observations to the period 2004-2013. Secondly, only the forecasted distance based adjacency matrix $\hat{\mathbf{D}}$ is used and not the forecasted stock prices.

Similarly to [2] a Multidimensional Scaling Technique, the Principal Coordinates Analysis is applied to $\hat{\mathbf{D}}$ with the aim of embedding the data in a space of lower dimensions while retaining the pairwise distances between the points as much as possible. The dimensionality reduction facilitates the classification of high-dimensional data, by mitigating the curse of dimensionality and other undesired properties of high-dimensional spaces. After having found the centering matrix $\mathbf{H} = \mathbf{I} - N^{-1}\mathbf{1}\mathbf{1}^T$, where \mathbf{I} is the $N \times N$ identity matrix, and $\mathbf{1}$ is a vector of N ones. The eigenvalue and eigenvectors of the matrix $\mathbf{B} = \mathbf{H} \left(-\frac{1}{2}\hat{\mathbf{D}}^2 \right) \mathbf{H}$ are found. The coordinates in the

lower-dimensional space are recorded in a matrix $\mathbf{X} = \mathbf{A}_s \mathbf{L}_s^{1/2}$. Where \mathbf{A}_s contains the eigenvectors corresponding to the s largest eigenvalues of \mathbf{B} , and $\mathbf{L}_s^{1/2}$ contains the square root of the s largest eigenvalues along the diagonal. Following [1] these points are embedded in a space of 6 dimension ($s = 6$). The 6th root of the product of the eigenvalues of $\mathbf{X}'\mathbf{X}$ defines the volume of the geometrical object composed by the embedded data. The volume is used as a reference for the identification of

abnormal periods. The volume expands whenever the cloud of points represents a situation of business as usual and the market space is similar to that of a random universe. On the other hand, in critical periods, the volume of the geometric object severely contracts, leading to the emergence of distorted shapes [2].

The investment strategy has been according modified to be applied to the S&P500 composite index. Now, the new signal is given by the difference of two subsequently predicted volumes $\Delta^{forc} = \widehat{V}^t - \widehat{V}^{t-1}$. Whenever this difference is negative the index is sold at price P^t and bought back the next trading day at price P^{t+1} . Otherwise, the index is bought at price P^t and sold back in $t + 1$ at price P^{t+1} . The cumulative returns are calculated as $\frac{P^t - P^{t+1}}{P^{t+1}}$ in the first case, and as $\frac{P^{t+1} - P^t}{P^t}$ in the second.

Fig. 4 shows the cumulative sum of the returns obtained by investing in the S&P500 composite index by following the differences in the predicted volumes. The simulations are obtained using the following parameters: $n_1 = 7, Z = 30, n_2 = 20$. As for the investment strategy based on stocks price predictions, also in this case, Fig. 4(a) shows that the predicted movements of stocks distances anticipates the market dynamic. Large down-turns are correctly anticipated as indicated by the cumulative returns illustrated in Fig. 4(b) near day 1000 (that corresponds to the initial period of the 2008 financial crisis). Fig.4(c), on the other hand, suggests that market up-turns, besides providing higher returns, are less severe than bust phases. The sum of the cumulative returns indeed has a smoother increasing behavior compared with the one obtained by correctly predicting market down-turns.

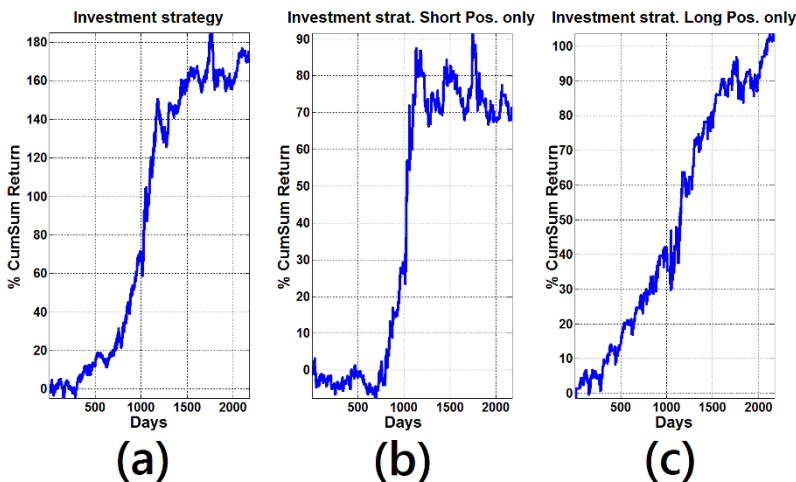


Fig. 4: Cumulative sum of returns obtained for the S&P500 composite index. Fig. 4(a) shows the performance of the investment strategy. Panels (b)-(c) display the cumulative returns obtained by investing only taking short or long positions respectively. This helps in quantifying how the correct forecasts of booms and burst phases affects the overall cumulative returns. The simulations are obtained using the following parameters: $n_1 = 7, Z = 30, n_2 = 20$.

Finally, Fig. 5 provides some robustness analysis. Since the parameter space is huge, the length of the tensor is kept fixed together with the parameter of the exponential smoothing while n_1 and n_2 take different values along the simulations. Each sub plot of Fig. 5 shows the cumulative returns obtained at the end of the time sample for different parameter values. In particular, the first row represents the end-of-sample cumulative returns obtained by averaging the cumulative performance of the method while forecasting the dynamic of the 388 stocks of the first dataset. In particular Fig. 5(a) refers to the composite investment strategy, encompassing both long and short positions. Panels 5(b)-(c), instead, differentiate between short and long positions respectively. The central row of Fig. 5 shows the returns obtained by following only the best signal (as emphasized also in Fig. 3), for the whole investment strategy (d) and for short (e) and long (f) positions respectively. The last row, on the other hand, provides the same results but looking at the performance obtained by the application of the modified method to the S&P500 composite index.

4 Discussion

The findings obtained by the application of this methodology have important consequences in the understanding of financial systems. As pointed out by the recent financial crisis indeed, financial systems are increasingly build on interdependencies and relationships that are difficult to predict and control. This work proposes a new dynamical approach to financial system and stresses the systemic importance of empirical signs that can be used to extend the knowledge of financial markets and complex systems in general. Predicting abrupt market down-turn, as a matter of fact, facilitate the design of policies that can reduce the hardness of financial crisis, plummeting the risk of global collapses of financial services by making economic networks more robust. The results suggest that tensor decompositions and multi-way analysis can effectively extract complex relationships from stock prices' time series opening new insights into large-scale collective decision making.

References

- [1] Araújo, T., Louçã, F.: The geometry of crashes. a measure of the dynamics of stock market crises. *Quantitative Finance* **7**(1), 63–74 (2007)
- [2] Arajo, T., Spelta, A.: Structural changes in cross-border liabilities: A multidimensional approach. *Physica A: Statistical Mechanics and its Applications* **394**, 277 – 287 (2014)
- [3] Bader, B.W., Kolda, T.G., et al.: Matlab tensor toolbox version 2.5. Available online (2012). URL <http://www.sandia.gov/~tgkolda/TensorToolbox/>
- [4] Carroll, J.D., Chang, J.J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of “eckart-young” decomposition. *Psychometrika* **35**(3), 283–319 (1970)
- [5] Dakos, V., van Nes, E.H., Donangelo, R., Fort, H., Scheffer, M.: Spatial correlation as leading indicator of catastrophic shifts. *Theoretical Ecology* **3**(3), 163–174 (2010)
- [6] Dakos, V., Scheffer, M., van Nes, E.H., Brovkin, V., Petoukhov, V., Held, H.: Slowing down as an early warning signal for abrupt climate change. *Proceedings of the National Academy of Sciences* **105**(38), 14,308–14,312 (2008)
- [7] Dunlavy, D.M., Kolda, T.G., Acar, E.: Temporal link prediction using matrix and tensor factorizations. *ACM Trans. Knowl. Discov. Data* **5**(2), 10:1–10:27 (2011)

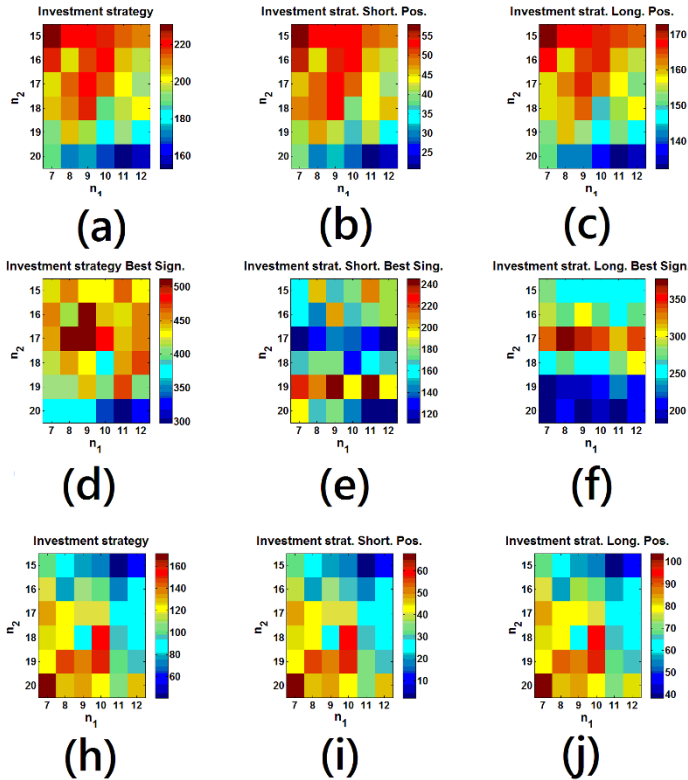


Fig. 5: Cumulative returns obtained at the end of the time sample for different values of n_1 and n_2 . Panel 5(a) represents the end-of-sample cumulative performance obtained by averaging the returns obtained by forecasting the dynamic of the 388 stocks of the first dataset. Panel 5(b) displays the end-of-sample returns by investing using only short positions and fig. 5(c) encompasses the results for the long positions investment strategy. Panel 5(d) shows the results for the investment strategy obtained by looking only at the best signal produced by the method together with the results for short (panel 5(e)) and long (panel 5(f)) investment strategy only. Finally, the last row provides the end-of-sample cumulative returns obtained by applying the modified method to the S&P500 composite index, for the whole investment strategy (panel 5(h)) and for short (panel 5(i)) and long (panel 5(j)) investment positions.

[8] For, C., Harshman, R.A.: Foundations of the parafac procedure: Models and conditions for an "explanatory" multimodal factor analysis by

[9] Gao, S., Denoyer, L., Gallinari, P.: Link pattern prediction with tensor decomposition in multi-relational networks. In: CIDM 2011 - IEEE Symposium on Computational Intelligence and Data Mining, pp. 333–340. IEEE, Paris, France (2011)

[10] Kéfi, S., Guttal, V., Brock, W.A., Carpenter, S.R., Ellison, A.M., Livina, V.N., Seekell, D.A., Scheffer, M., van Nes, E.H., Dakos, V.: Early warning signals of ecological transitions: Methods for spatial patterns. PLoS ONE 9(3), 1–13 (2014)

- [11] Kolda, T.G., Bader, B.W., Kenny, J.P.: Higher-order web link analysis using multilinear algebra. In: Proceedings of the Fifth IEEE International Conference on Data Mining, ICDM '05, pp. 242–249. IEEE Computer Society, Washington, DC, USA (2005)
- [12] Lü, L., Zhou, T.: Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications* **390**(6), 1150 – 1170 (2011)
- [13] Moon, H., Lu, T.C.: Network catastrophe: Self-organized patterns reveal both the instability and the structure of complex networks. *Scientific reports* **5** (2015)
- [14] Onnela, J.P., Chakraborti, A., Kaski, K., Kertiész, J.: Dynamic asset trees and portfolio analysis. *The European Physical Journal B-Condensed Matter and Complex Systems* **30**(3), 285–288 (2002)
- [15] Preis, T., Kenett, D.Y., Stanley, H.E., Helbing, D., Ben-Jacob, E.: Quantifying the behavior of stock correlations under market stress. *Scientific reports* **2** (2012)
- [16] Preis, T., Moat, H.S., Stanley, H.E.: Quantifying trading behavior in financial markets using google trends. *Scientific reports* **3** (2013)
- [17] Preis, T., Schneider, J.J., Stanley, H.E.: Switching processes in financial markets. *Proceedings of the National Academy of Sciences* **108**(19), 7674–7678 (2011)
- [18] Scheffer, M., Bascompte, J., Brock, W.A., Brovkin, V., Carpenter, S.R., Dakos, V., Held, H., Van Nes, E.H., Rietkerk, M., Sugihara, G.: Early-warning signals for critical transitions. *Nature* **461**(7260), 53–59 (2009)
- [19] Tumminello, M., Di Matteo, T., Aste, T., Mantegna, R.: Correlation based networks of equity returns sampled at different time horizons. *The European Physical Journal B* **55**(2), 209–217 (2007)
- [20] Xu, Y., Yin, W.: A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences* **6**(3), 1758–1789 (2013)

Who buys what, where: Reconstruction of the international trade flows by commodity and industry

Yuichi Ikeda and Tsutomu Watanabe

Abstract We developed a model to reconstruct the international trade network by considering both commodities and industry sectors in order to study the effects of reduced trade costs. First, we estimated trade costs to reproduce WIOD and NBER-UN data. Using these costs, we estimated the trade costs of sector specific trade by types of commodities. We successfully reconstructed sector-specific trade for each types of commodities by maximizing the configuration entropy with the estimated costs. In WIOD, trade is actively conducted between the same industry sectors. On the other hand, in NBER-UN, trade is actively conducted between neighboring countries. This seems like a contradiction. We conducted community analysis for the reconstructed sector-specific trade network by type of commodities. The community analysis showed that products are actively traded among same industry sectors in neighboring countries. Therefore the observed features of the community structure for WIOD and NBER-UN are complementary.

1 Introduction

In the era of economic globalization, most national economies are linked by international trade, which in turn consequently forms a complex global economic network. It is believed that greater economic growth can be achieved through free trade based on the establishment of Free Trade Agreements (FTAs) and Economic Partnership Agreements (EPAs). In the last years, many researchers have studies international trade from a perspective of network science [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. However, there is limitation to the resolution of the currently available trade data. For instance, NBER-UN records trade amounts between bilateral countries without

Y. Ikeda (e-mail: ikeda.yuichi.2w@kyoto-u.ac.jp)✉
Graduate School of Advanced Integrated Studies in Human Survivability, Kyoto University

T. Watanabe e-mail: watanabe@e.u-tokyo.ac.jp
Graduate School of Economics, University of Tokyo

industry sector information for each type of commodities [13], and the World Input-Output Database (WIOD) records sector-specific trade amount without commodities information [14]. This limited resolution makes it difficult to analyze community structures in detail and systematically assess the effects of reduced trade tariffs and trade barriers.

In this paper, we reconstruct the sector-specific trade network for each type of commodities by maximizing the configuration entropy based on the local information about the inward and outward flow of trade. The reconstruction of interbank networks from local information has been studied intensively [15, 16, 17]. But these studies intend to reproduce the average nearest degree, the average nearest strength, and the expected weight for various weighted networks. Our goal is to reconstruct an international trade network by considering both commodities and industry sectors in order to systematically study the effects of reduced trade costs, such as trade tariffs and trade barriers.

This paper is organized as follows: Section 2 describes the model of network reconstruction, and Section 3 explains the existing trade data. Section 4 shows results of cost estimation, and finally Section 5 explains the identified community structure for the reconstructed international trade network. Section 6 provides a summary of the points presented in this paper.

2 Model of Network Reconstruction

We reconstruct the international trade network by considering both commodities and industry sectors in order to systematically study the effects of reduced trade costs. For this reason, the estimation of trade costs is indispensable. In this section, we describe our network reconstruction model.

2.1 Outline of Network Reconstruction Model

We reconstruct the sector-specific trade network for each type of commodities by maximizing the configuration entropy using existing international trade data: NBER-UN and WIOD. These two types of existing data will be explained in the following section.

The outline of our model of network reconstruction is as follows:

1. We estimate trade cost $C_{AB}^{(G)}$ between country A and B for commodities (G) to reproduce trade amount data NBER-UN $T_{AB}^{(G)}$ by maximizing the configuration entropy with given strengths $D_A^{(G)}$ and $O_B^{(G)}$.
2. We estimate trade cost $C_{A\alpha B\beta}$ to reproduce trade amount data WIOD $T_{A\alpha B\beta}$ between industry sector α in country A and industry sector β in country B by maximizing the configuration entropy with given strengths $D_{A\alpha}$ and $O_{B\beta}$.

3. We obtain an analytic formulae to calculate trade cost $C_{A\alpha B\beta}^{(G)}$ using costs estimated above: $C_{AB}^{(G)}$ and $C_{A\alpha B\beta}$.
4. We calculate the trade cost $C_{A\alpha B\beta}^{(G)}$ analytically and estimate the sector-specific trade for each type of commodities $T_{A\alpha B\beta}^{(G)}$ by maximizing the configuration entropy with given strengths $T_{AB}^{(G)}$ and $T_{A\alpha B\beta}$.

2.2 Maximization of the Configuration Entropy

A model that calculates the amount of traffic flow based on the local information for total outflow and inflow by maximizing the configuration entropy has been proposed [18]. We apply this model for our purpose. Suppose that the total amount of export O_i from country i , total amount of import D_j to country j , and trade cost C_{ij} from country i to country j are given: $O_i = \sum_j T_{ij}$, $D_j = \sum_i T_{ij}$, and $C = \sum_{ij} T_{ij} C_{ij}$. The formulation of export T_{ij} from country i to j is obtained by maximizing the configuration entropy $S = \log W = \log \left((\sum_{ij} T_{ij})! / \prod_{ij} T_{ij}! \right)$ with the constraints using the Lagrange multiplier method. As a result, we obtain the closed relationship for export T_{ij} as follows:

$$T_{ij} = A_i B_j O_i D_j \exp(-\beta C_{ij}), \tag{1}$$

$$A_i = \left[\sum_j B_j D_j \exp(-\beta C_{ij}) \right]^{-1}, \tag{2}$$

$$B_j = \left[\sum_i A_i O_i \exp(-\beta C_{ij}) \right]^{-1}. \tag{3}$$

Here β is a multiplier that signifies the constraint for total trade cost C_{ij} . Coefficients A_i and B_j are calculated iteratively from the appropriate initial values.

2.3 Algorithm of Cost Estimation

Figure 1 shows the algorithm of cost estimation. The trade cost is estimated using simulated annealing [19] to reproduce the actual trade data. The simulated annealing takes a long time to compute, but shows a reasonably good convergence of the cost estimation. The cooling schedule of temperature T is given by $T_n = (1 - 0.003)^n$. Here n is the number of iteration step n . At each temperature, the calculation is repeated using equilibrium samples. The root mean square error of calculated trade $RMS_n = \sqrt{\sum_{ij} ((T_{ij}^{cal} - T_{ij}) / T_{ij})^2} / N$ is calculated at each iteration step. Here T_{ij}^{cal} , T_{ij} , and N are calculated trade for a given cost, actual trade, and the number of combination of countries, respectively. If $\Delta RMS_n = RMS_n - RMS_{n-1}$ is negative,

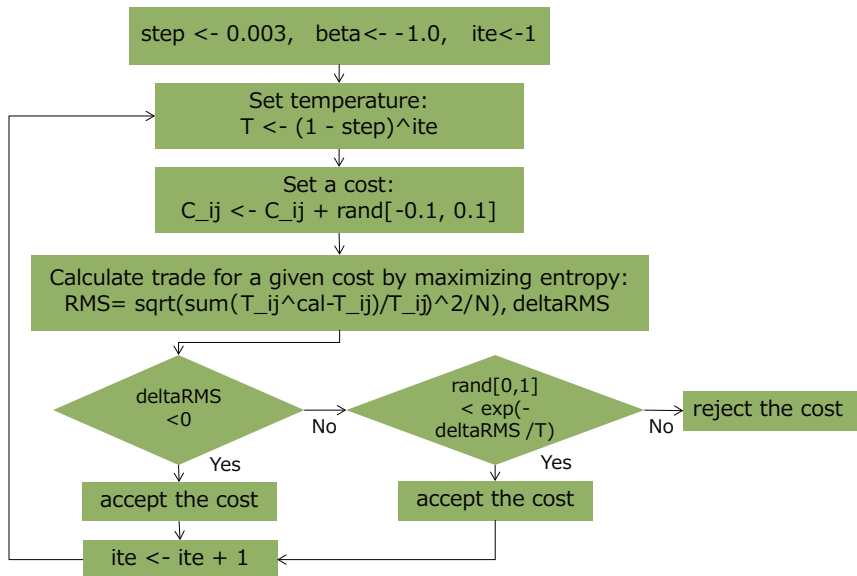


Fig. 1: The algorithm of cost estimation. Trade cost is estimated using simulated annealing to reproduce the actual trade data.

we accept cost C_{ij} , but if ΔRMS_n is positive, the acceptance of cost is determined stochastically depending on the temperature.

2.4 Sector-Specific Cost by Commodities

The analytical formula of sector-specific cost $C_{A\alpha B\beta}^{(G)}$ by type of commodities is obtained as a weighted average of the trade costs for WIOD and NBER-UN: $C_{A\alpha B\beta}$ and $C_{AB}^{(G)}$. We have three identities:

$$\sum_{\alpha\beta} T_{A\alpha B\beta} C_{A\alpha B\beta} = \sum_G T_{AB}^{(G)} C_{AB}^{(G)} = \sum_{\alpha\beta G} T_{A\alpha B\beta}^{(G)} C_{A\alpha B\beta}^{(G)} = T_{AB} C_{AB}, \tag{4}$$

$$T_{A\alpha B\beta} C_{A\alpha B\beta} = \sum_G T_{A\alpha B\beta}^{(G)} C_{A\alpha B\beta}^{(G)}, \tag{5}$$

$$T_{AB}^{(G)} C_{AB}^{(G)} = \sum_{\alpha\beta} T_{A\alpha B\beta}^{(G)} C_{A\alpha B\beta}^{(G)}. \tag{6}$$

Using these identities, we write trade cost $C_{A\alpha B\beta}^{(G)}$ as a weighted average of $C_{A\alpha B\beta}$ and $C_{AB}^{(G)}$.

$$C_{A\alpha B\beta}^{(G)} = \frac{1}{2} \left(u_G \frac{T_{A\alpha B\beta}}{T_{A\alpha B\beta}^{(G)}} C_{A\alpha B\beta} + v_{\alpha\beta} \frac{T_{AB}^{(G)}}{T_{A\alpha B\beta}^{(G)}} C_{AB}^{(G)} \right), \quad (7)$$

$$u_G = \frac{C_{AB}^{(G)}}{C_{AB}}, \quad (8)$$

$$v_{\alpha\beta} = \frac{C_{A\alpha B\beta}}{C_{AB}}. \quad (9)$$

We obtain the following analytical formula of the sector-specific cost by type of commodities as an approximation:

$$C_{A\alpha B\beta}^{(G)} \cong \frac{1}{2} \left(u_G G C_{A\alpha B\beta} + v_{\alpha\beta} S^2 C_{AB}^{(G)} \right). \quad (10)$$

2.5 Sector-Specific Trade by Type of Commodities

Once $C_{A\alpha B\beta}^{(G)}$ is obtained, we estimate $T_{A\alpha B\beta}^{(G)}$ based on the given local information $T_{AB}^{(G)}$ and $T_{A\alpha B\beta}$ by maximizing entropy iteratively, in the same manner as before.

$$T_{A\alpha B\beta}^{(G)} = \tilde{A}_{AB}^{(G)} \tilde{B}_{A\alpha B\beta} T_{AB}^{(G)} T_{A\alpha B\beta} \exp \left(-\beta C_{A\alpha B\beta}^{(G)} \right), \quad (11)$$

$$\tilde{A}_{AB}^{(G)} = \left[\sum_{\alpha\beta} \tilde{B}_{A\alpha B\beta} T_{A\alpha B\beta} \exp \left(-\beta C_{A\alpha B\beta}^{(G)} \right) \right]^{-1}, \quad (12)$$

$$\tilde{B}_{A\alpha B\beta} = \left[\sum_G \tilde{A}_{AB}^{(G)} T_{AB}^{(G)} \exp \left(-\beta C_{A\alpha B\beta}^{(G)} \right) \right]^{-1}. \quad (13)$$

3 Trade Data

We used bilateral trade data between countries for each type of commodities NBER-UN and sector-specific trade data WIOD at year 2000. Table 1 shows the list of commodities for NBER-UN. Table 2 shows the list of countries for NBER-UN and WIOD. Table 3 shows the list of industry sectors for WIOD. Here the number of commodities G is 10, the number of countries N is 31, and the number of industry sectors S is 35.

Figure 2 shows that the relationship between NBER-UN and WIOD for the total amount of exports and imports of 31 countries. We note that WIOD is about 50% to 60% of NBER-UN for both exports and imports. We assume that the difference between the two databases comes from the lack of a consumer sector in WIOD.

Table 1: Commodities for NBER-UN

Symbol	Description
g0	<i>FOOD AND LIVE ANIMALS CHIEFLY FOR FOOD</i>
g1	<i>BEVERAGES AND TOBACCO</i>
g2	<i>CRUDE MATERIALS, INEDIBLE, EXCEPT FUELS</i>
g3	<i>MINERAL FUELS, LUBRICANTS AND RELATED MATERIALS</i>
g4	<i>ANIMAL AND VEGETABLE OILS, FATS AND WAXES</i>
g5	<i>CHEMICALS AND RELATED PRODUCTS, N.E.S.</i>
g6	<i>MANUFACTURED GOODS CLASSIFIED CHIEFLY BY MATERIAL</i>
g7	<i>MACHINERY AND TRANSPORT EQUIPMENT</i>
g8	<i>MISCELLANEOUS MANUFACTURED ARTICLES</i>
g9	<i>COMMODITIES & TRANS. NOT CLASSIFIED ELSEWHERE</i>

Table 2: Countries for NBER-UN and WIOD

Symbol	Description	Symbol	Description	Symbol	Description	Symbol	Description
c1	<i>Australia</i>	c2	<i>Austria</i>	c3	<i>Bulgaria</i>	c4	<i>Brazil</i>
c5	<i>Canada</i>	c6	<i>China</i>	c7	<i>CzechRep</i>	c8	<i>Germany</i>
c9	<i>Denmark</i>	c10	<i>Spain</i>	c11	<i>Finland</i>	c12	<i>France</i>
c13	<i>UK</i>	c14	<i>Greece</i>	c15	<i>Hungary</i>	c16	<i>Indonesia</i>
c17	<i>Ireland</i>	c18	<i>Italy</i>	c19	<i>Japan</i>	c20	<i>KoreaRep</i>
c21	<i>Mexico</i>	c22	<i>Netherlands</i>	c23	<i>Poland</i>	c24	<i>Portugal</i>
c25	<i>Romania</i>	c26	<i>RussianFed</i>	c27	<i>Slovakia</i>	c28	<i>Slovenia</i>
c29	<i>Sweden</i>	c30	<i>Turkey</i>	c31	<i>USA</i>		

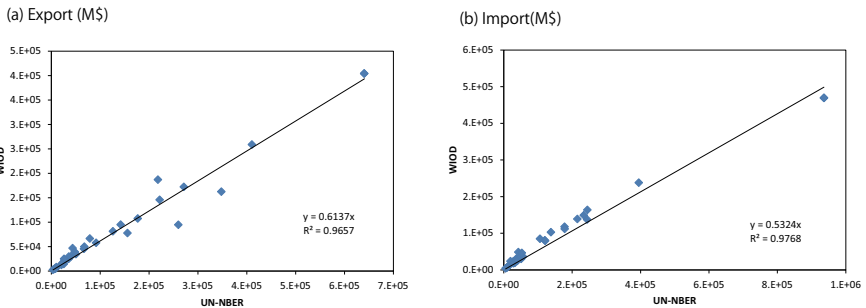


Fig. 2: The relationship between NBER-UN and WIOD for the total amount of exports and imports of 31 countries.

4 Cost Estimation

In this section, first we estimated $C_{AB}^{(G)}$ to reproduce the trade amount data for NBER-UN. Then, we estimated $C_{A\alpha B\beta}$ to reproduce the trade amount data for WIOD. Finally,

Table 3: Industry Sectors for WIOD

Symbol	Description
s1	<i>Agriculture, Hunting, Forestry and Fishing</i>
s2	<i>Mining and Quarrying</i>
s3	<i>Food, Beverages and Tobacco</i>
s4	<i>Textiles and Textile Products</i>
s5	<i>Leather, Leather and Footwear</i>
s6	<i>Wood and Products of Wood and Cork</i>
s7	<i>Pulp, Paper, Paper, Printing and Publishing</i>
s8	<i>Coke, Refined Petroleum and Nuclear Fuel</i>
s9	<i>Chemicals and Chemical Products</i>
s10	<i>Rubber and Plastics</i>
s11	<i>Other Non-Metallic Mineral</i>
s12	<i>Basic Metals and Fabricated Metal</i>
s13	<i>Machinery, Nec</i>
s14	<i>Electrical and Optical Equipment</i>
s15	<i>Transport Equipment</i>
s16	<i>Manufacturing, Nec; Recycling</i>
s17	<i>Electricity, Gas and Water Supply</i>
s18	<i>Construction</i>
s19	<i>Sale, Maintenance and Repair of Motor Vehicles and Motorcycles; Retail Sale of Fuel</i>
s20	<i>Wholesale Trade and Commission Trade, Except of Motor Vehicles and Motorcycles</i>
s21	<i>Retail Trade, Except of Motor Vehicles and Motorcycles; Repair of Household Goods</i>
s22	<i>Hotels and Restaurants</i>
s23	<i>Inland Transport</i>
s24	<i>Water Transport</i>
s25	<i>Air Transport</i>
s26	<i>Other Supporting and Auxiliary Transport Activities; Activities of Travel Agencies</i>
s27	<i>Post and Telecommunications</i>
s28	<i>Financial Intermediation</i>
s29	<i>Real Estate Activities</i>
s30	<i>Renting of M&Eq and Other Business Activities</i>
s31	<i>Public Admin and Defence; Compulsory Social Security</i>
s32	<i>Education</i>
s33	<i>Health and Social Work</i>
s34	<i>Other Community, Social and Personal Services</i>
s35	<i>Private Households with Employed Persons</i>

we calculated $C_{A\alpha B\beta}^{(G)}$ using the analytic formula in Eq. (10) as a weighted average of $C_{A\alpha B\beta}$ and $C_{AB}^{(G)}$.

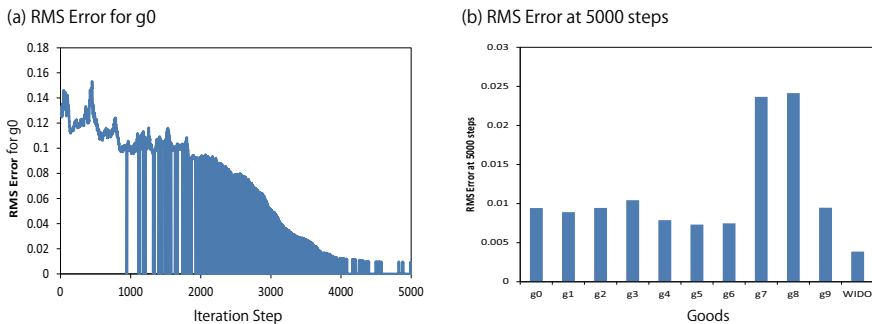


Fig. 3: RMS error for g0: food and live animals chiefly, and RMS errors at 5000 steps for various commodities and WIOD.

4.1 Trade Cost of WIOD

The left panel of Fig. 3 shows that the convergence of the RMS errors for g0: *FOOD AND LIVE ANIMALS CHIEFLY FOR FOOD* and the right panel shows the RMS errors at 5000 steps for various type of commodities and WIOD. The error for each trade cost is 0.5% to 2% for all commodities and WIOD. Figure 4 shows the comparison of (a) actual trade $T_{A\alpha B\beta}$ and (b) calculated trade $T_{A\alpha B\beta}$ using estimated cost $C_{A\alpha B\beta}$. The agreement between two types of trade is quite good.

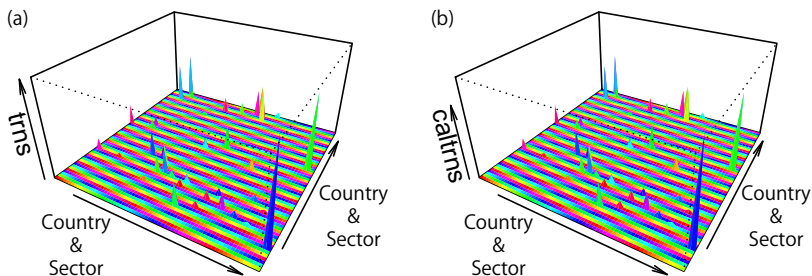


Fig. 4: Comparison of actual trade $T_{A\alpha B\beta}$ and calculated trade $T_{A\alpha B\beta}$ using estimated cost $C_{A\alpha B\beta}$.

4.2 Trade Cost of NBER-UN

Figure 5 shows the comparison of (a) actual trade $T_{AB}^{(G)}$ and (b) calculated trade $T_{AB}^{(G)}$ using estimated cost $C_{AB}^{(G)}$ for commodity g7: *MACHINERY AND TRANSPORT*

EQUIPMENT. The agreement between these two types of trade is once again quite good.

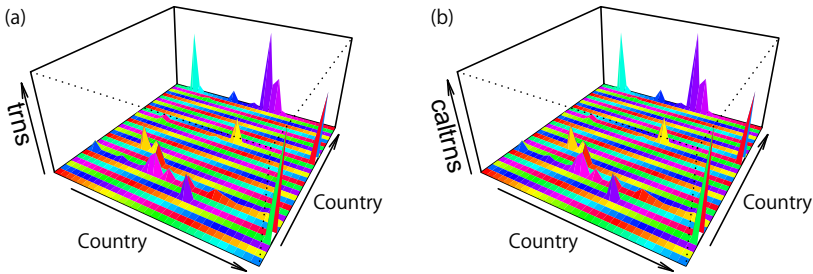


Fig. 5: Comparison of actual trade $T_{AB}^{(G)}$ and calculated trade $T_{AB}^{(G)}$ using estimated cost $C_{AB}^{(G)}$ for commodity g7.

4.3 Estimated Sector-Specific Cost by Type of Commodities

Sector-specific cost by type of commodities was estimated using Eq. (10). The estimated cost $C_{A\alpha\beta}^{(G)}$ for commodity g5: *CHEMICALS AND RELATED PRODUCTS, N.E.S.* and g7: *MACHINERY AND TRANSPORT EQUIPMENT* are shown in Figs. 6 and 7, respectively. Note that we have common characteristics for both g5 and g7. For example, import costs in the german transport equipment industry are very high compared with other industry sectors of various countries. In the US, import costs are higher than export costs for many industries. On the other hand, in Japan, export costs are higher than import costs for some industries.

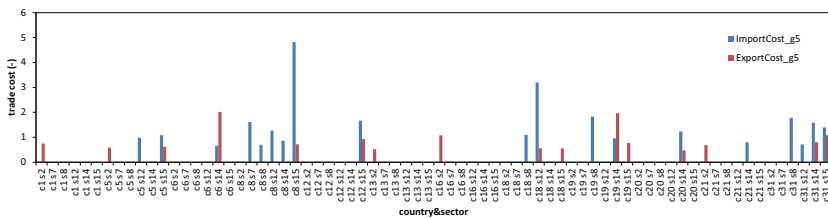


Fig. 6: Estimated trade cost $C_{A\alpha\beta}^{(G)}$ for commodity g5: *CHEMICALS AND RELATED PRODUCTS, N.E.S.*.

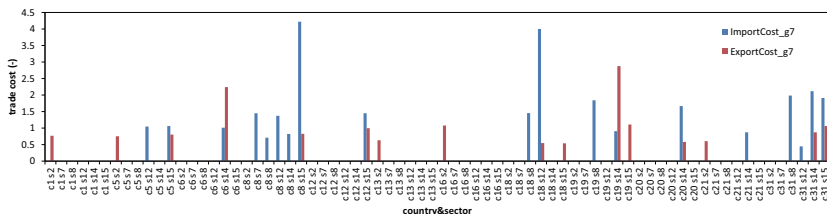


Fig. 7: Estimated trade cost $C_{A\alpha B\beta}^{(G)}$ for commodity g7: *MACHINERY AND TRANSPORT EQUIPMENT*.

5 Reconstructed Sector-Specific Trade Network by Type of Commodities

$T_{A\alpha B\beta}^{(G)}$ provides a weight for links of the sector-specific trade network by each type of commodities. For the reconstructed international trade network, we identify a community structure that corresponds to economic clusters linked by the trade of various type of commodities. In past analysis of the sector-specific trade network (WIOD), we obtained communities consisting of the same industry sector across countries [20, 21, 22]. In this section, we describe the characteristics of the community structure identified for the reconstructed sector-specific trade network by type of commodities.

5.1 Community Structure in WIOD

The community structure was identified by maximizing the modularity for WIOD. The identified community shows that the international trade is actively conducted between the same or similar industry sectors [22], but it is not know which commodities are traded. We note that a defect has been pointed out for the null model used in the definition of the modularity for weighted networks [15]. We conducted community analysis using map equation [23] for WIOD to confirm the community structure identified by modularity maximization. We confirmed that international trade is actively conducted between the same or similar industry sectors. The largest community consists of industry sector: *Renting of M&Eq, Financial Intermediation*, the second is industry sector: *Chemical Products*, the third is industry sector: *Basic Metals and Fabricated Metal*, the fourth is industry sector: *Mining and Quarrying*, the fifth is industry sector: *Electrical and Optical Equipment*, and the sixth is industry sector: *Transport Equipment*.

5.2 Community Structure in NBER-UN

Community analysis for NBER-UN shows that international trade is actively conducted between neighboring countries, but industry sectors in which trade is conducted are not known. For example, we found five communities for g5: *CHEMICALS AND RELATED PRODUCTS, N.E.S.*. The largest community is Europe, consisting of Austria, Bulgaria, the Czech Republic, Germany, Spain, Finland, France, the UK, Hungary, Italy, the Netherlands, Poland, Portugal, the Russian Federation, Slovakia, and Slovenia. The second is South & North America, consisting of Brazil, Canada, Ireland, Mexico, and the USA. The third is Asia, consisting of Australia, China, Indonesia, Japan, and Korea Republic. The fourth is West Asia & East Europe, consisting of Greece, Romania, and Turkey. The fifth is North Europe, consisting of Denmark and Sweden.

5.3 Community Structure in Reconstructed Sector-Specific Trade Network by Commodities

Community analysis of the sector-specific trade network (WIOD) shows that international trade is actively conducted between the same or similar industry sectors.

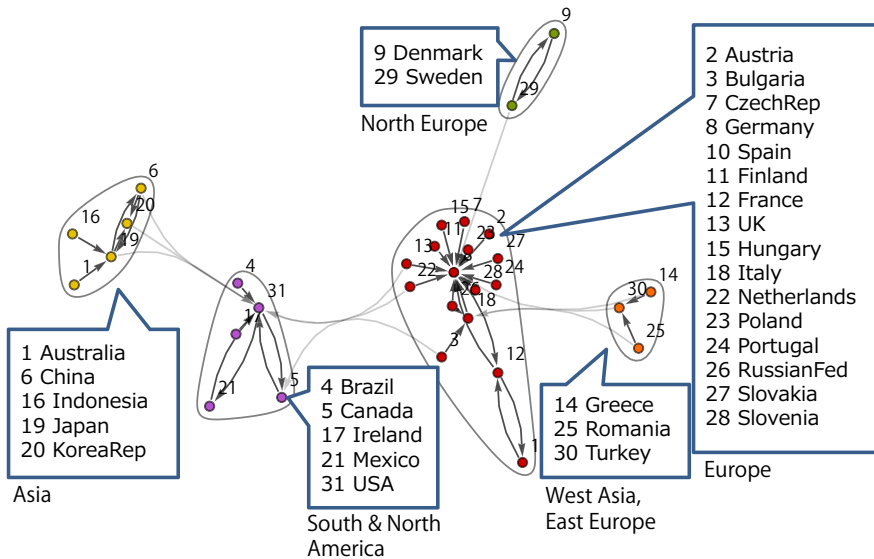


Fig. 8: Community Structure in NBER-UN for commodity g5. International trade is actively conducted between neighboring countries, but industry sectors in which trade is conducted are not known.

On the other hand, community analysis of the trade network for a specific type of commodities (NBER-UN) shows that international trade is actively conducted between neighboring countries. At first glance, these results seem to be contradictory. What do these results really mean?

We conducted community analysis for the reconstructed sector-specific trade network by type of commodity g5: *CHEMICALS AND RELATED PRODUCTS, N.E.S.*. The identified community structure is shown in Fig. 9. The largest community corresponds to Europe, and all nodes in this community are in the Transport Equipment industry sector. The second largest community corresponds to South & North America, and all nodes are in the Electrical and Optical Equipment industry sector. In a similar way, the third largest community corresponds to West Asia & East Europe, and all nodes are in the Basic Metals and Fabricated Metal industry sector. Analysis showed that products are actively traded between the same industry sectors in neighboring countries. Therefore, we can say that the observed features of the community structure for WIOD and NBER-UN are not contradictory but rather that they are complementary.

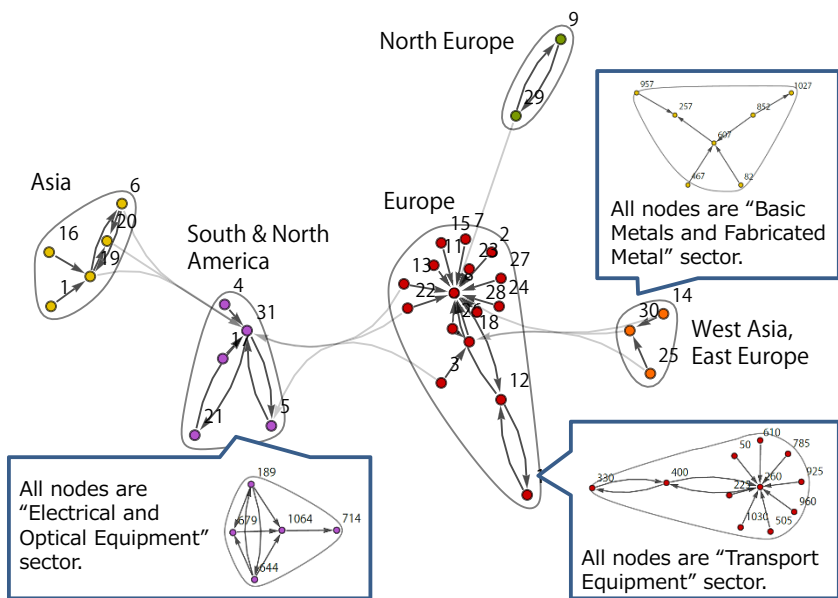


Fig. 9: Community Structure in Reconstructed Sector-Specific Trade Network for commodity g5. Three insets show the identified community structure for the reconstructed sector-specific trade network. Commodity g5 are actively traded between the same industry sectors in neighboring countries.

6 Summary

We developed a model to reconstruct the international trade network by considering both commodities and industry sectors in order to study the effects of reduction of various trade costs. First, we estimated the trade cost to reproduce WIOD and NBER-UN data. Using these costs, we estimated the trade cost of sector specific trade by type of commodities. We successfully reconstructed sector-specific trade for each type of commodities by maximizing the configuration entropy with the estimated cost.

In WIOD, trade is actively conducted between the same industry sectors. On the other hand, in NBER-UN, trade is actively conducted between neighboring countries. This seems like a contradiction. We conducted community analysis for the reconstructed sector-specific trade network by type of commodity g_5 . The community analysis showed that products are actively traded between the same industry sectors in neighboring countries. The observed features of the community structure for WIOD and NBER-UN are complementary.

In future studies, we intend to analyze the effect of reduced trade tariffs and trade barriers. For instance, the Trans-Pacific Partnership (TPP) is expected to achieve a high-level of free trade in the Asia-Pacific region, which accounts for more than 40% of the world's GDP. Trade costs are estimated at 170% of the price of commodities. The breakdown in transportation costs is 21%, and the rest is trade tariffs and trade barriers [24]. We will discuss the effect of reduced trade tariffs and trade barriers on the change in the community structure of the international trade network. This will enable us to arrive at better understanding of international trade after the TPP agreement goes into effect.

Acknowledgements The present study was supported in part by the Ministry of Education, Science, Sports, and Culture, Grants-in-Aid for Scientific Research (C), Grant No. 26350422 (2014-16). The authors are grateful for helpful discussion and comments by H. Iyetomi (Niigata University), T. Mizuno (National Institute of Informatics), and T. Ohnishi (University of Tokyo).

References

- [1] X. Li, Y. Jin, and G. Chen, Complexity and synchronization of the world trade web, *Physica A: Statistical Mechanics and its Applications*, 328, 1, pp. 287–296, 2003.
- [2] D. Garlaschelli and M. I. Loffredo, Structure and evolution of the world trade network, *Physica A* 355, pp. 138144, 2005.
- [3] C. A. Hidalgo, B. Klinger, A-L, Barabási, and R. Hausmann, The product space conditions the development of nations, *Science*, 317, 5837, pp. 482–487, 2007.
- [4] K. Bhattacharya, G. Mukherjee, J. Saramäki, K. Kaski, and S. S. Manna, The international trade network: weighted network analysis and modelling, *Journal of Statistical Mechanics: Theory and Experiment*, 2008, 02, P02002, 2008.
- [5] G. Fagiolo, J. Reyes, and S. Schiavo, World-trade web: Topological properties, dynamics, and evolution, *Physical Review E*, 79, 3, p. 036115, 2009.
- [6] L. De Benedictis and L. Tajoli, The world trade network, *The World Economy*, 34, 8, pp.1417–1454, 2011.
- [7] M. Barigozzi, G. Fagiolo, and G. Mangioni, Identifying the community structure of the international-trade multi-network, *Physica A* 390, pp.20512066, 2011.

- [8] K. Lee, J. Yang, G. Kim, J. Lee, K. Goh, and I. Kim, Impact of the topology of global macroeconomic network on the spreading of economic crises, *PloS one*, 6, 3, e18443, 2011.
- [9] M. Duenas and G. Fagiolo, Modeling the International-Trade Network: a gravity approach, *Journal of Economic Interaction and Coordination*, 8, 1, pp.155–178, 2013.
- [10] T. Deguchi, K. Takahashi, H. Takayasu, and M. Takayasu, Hubs and authorities in the world trade network using a Weighted HITS algorithm, *PloS one*, 9, 7, e100338, 2014.
- [11] L. Ermann and D. L. Shepelyansky, Google matrix analysis of the multiproduct world trade network, *The European Physical Journal B*, 88, 4, pp. 1–19, 2015.
- [12] F. Saracco, R. Di Clemente, A. Gabrielli, and T. Squartini, Randomizing bipartite networks: the case of the World Trade Web, *arXiv preprint arXiv:1503.05098*, 2015.
- [13] R. C. Feenstra et al, *WORLD TRADE FLOWS: 1962-2000*, NBER Working Paper 11040, 2005.
- [14] M. P. Timmer et al, *The World Input-Output Database : Contents, Sources and Methods*, WIOD Working Paper Number: 10, 2012.
- [15] R. Mastrandrea, T. Squartini, G. Fagiolo, D. Garlaschelli, *New J. Phys.* 16, 043022, 2014.
- [16] G. Cimini, T. Squartini, A. Gabrielli, D. Garlaschelli, *Phys. Rev. E* 92, 040802(R), 2015.
- [17] G. Cimini, T. Squartini, D. Garlaschelli, A. Gabrielli, *Scientific Reports* 5, 15758, 2015.
- [18] A.G. Wilson, *A Statistical Theory of Spatial Distribution Models*, Transportation Research, Vol.1, pp.253-269, 1967.
- [19] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, Optimization by Simulated Annealing, *Science* 220 (4598), pp.671680, 1983.
- [20] Y. Ikeda, H. Iyetomi, T. Mizuno, T. Ohnishi, T. Watanabe, Community Structure and Dynamics of the Industry Sector-Specific International-Trade-Network, in *Signal-Image Technology and Internet-Based Systems (SITIS)*, 2014, pp.456-461, 23-27 Nov. 2014. doi: 10.1109/SITIS.2014.67
- [21] Y. Ikeda, H. Aoyama, Y. Sakamoto, Community Dynamics and Controllability of G7 Global Production Network, in *Signal-Image Technology and Internet-Based Systems (SITIS)*, 2015, pp.391-397, 23-27 Nov. 2015. DOI 10.1109/SITIS.2015.28
- [22] Y. Ikeda, H. Aoyama, H. Iyetomi, T. Mizuno, T. Ohnishi, Y. Sakamoto, T. Watanabe, "Econophysics Point of View of Trade Liberalization: Community dynamics, synchronization, and controllability as example of collective motions", *DPRIETI Discussion Paper Series 16-E-026*, 2016.
- [23] M. Rosvall and C. T. Bergstrom, Maps of random walks on complex networks reveal community structure", *Proceedings of the National Academy of Sciences*, 105, 4, pp. 1118-1123, 2008.
- [24] J. E. Anderson and E. van Wincoop, Trade Costs, *Journal of Economic Literature*, Vol. 42, No. 3., pp. 691-751, 2004.

Network of Networks: A Meta-model for Simulated Financial Markets

Talal Alsulaiman and Khaldoun Khashanah

Abstract We investigate the properties of a calibrated network structure of an agent-based model for a simulated financial market. A meta-model of a network of networks is introduced to capture the simulated market structure. The agent-based model consists of heterogeneous agents characterized by two-dimensional attributes that are investment behavior and investment strategy. The resulting groups of agents are viewed as subnetworks giving rise to a network of networks (NoN). The aggregation of activities of agents in a subnetwork trickles up to shape the aggregate activities of the NoN. The objective of introducing the NoN is to provide a testbed for complex models of simulated markets. Furthermore, we investigate the emergence of the market patterns in terms of prices, moments of returns, market capital, and wealth distributions. The investigation was performed for fully connected homogeneous agents. The results show a significant difference in the market emergence behaviors in terms of prices and returns, however, the market capitalization stays close to the calibrated financial market. Also, the deviation of wealth distributions was less than those in the heterogeneous market.

1 Introduction

This paper introduces the concept of a meta-model for a network of networks (NoN) that appears in the course of creating an agent-based model for a market. The meta-model categorizes market participants based on their strategies coupled with their investment behavior. Strategies in real markets can vary greatly among agents but we restrict them to four possible categories of strategies: zero-intelligence, fundamental strategy, momentum trend-following strategy, and adaptive trading strategy using the artificial neural network (ANN) algorithm. Investment behavior as well can have an infinite spectrum but for our meta-model, we account for agents that can be risk

Talal Alsulaiman (e-mail: talal.alsulaiman@gmail.com)✉ · Khaldoun Khashanah
Financial Engineering Division, Stevens Institute of Technology, Hoboken, NJ

averse or loss occupied with overconfidence or conservative biases. The agents may interact with each other by sharing market sentiments through a structured scale-free network.

The meta-model introduced in this paper consists of a two-dimensional NoN in the sense that each category of strategy agents build up a network so that four subnetworks exist in a natural way in the dimension of strategy. On the other hand, there are six subnetworks classified by behavior. There is a total of twenty-four subnetworks when considering the network as a two-dimensional attribute network with strategy and behavior as the two dimensions. The quantitative rules of interaction among agents in a subnetwork give rise to quantitative inter-network interactions. Those interactions summarize the economic values exchanged by the agents individually, which trickles up to the level of the subnetwork as agent interactions are aggregated. The aggregation of the subnetworks interactions in the NoN gives rise to indicators of the state of the entire NoN. Thus giving us a way to model multi-scale structural change in large financial and economic networks using agent-based modeling. The granularity of the type of agent in the meta-model takes into consideration why the model is postulated. Alternatively, the granularity of the available data may dictate the level of the agent in the design. For example, in our illustrative meta-model, the basic level of an agent represents a trader while the measured response of interactions is taken at the market level in terms of a NoN price of the asset. For tractability, the measured response, in this case, is simply the price of a single asset traded in that market with an interest rate instrument. The price can be viewed as the aggregation of a "vote" by various two-dimensional subnetworks, which in this case total to twenty-four subnetworks. Each subnetwork has its own perception of a value that should be assigned to the asset due to its two-dimensional subnetwork classification, i.e. its strategy and behavior. The price at a given time is determined by the aggregate forces of heterogeneous agents supply and demand represented by the bids and offers from all subnetworks simultaneously. Once a transaction is consummated, a new (local) equilibrium price is posted and a new information feedback loop triggers agents to examine a new position.

Since our meta-model allows for agent change, the subnetworks can undergo migrations of agents from one subnetwork to another based on their observations of successful strategies. As a result, the meta-model can provide a cross-sectional view of NoN evolution in time. The applications of a dynamic structural change in large networks should be a topic of interest for socio-financial, economic studies but also for marketing segmentation as well as political decision support systems.

This paper investigates the effect of the network assortativity on a meta-model that appears in the course of creating an agent-based model for a market

The outline of the paper is as follows, in section 2 we survey the related studies, in 3 we provide a description of the meta-model for the financial market. Section 4 investigates the properties of the calibrated network in the meta-model. In section 5 we implement a set of four experiments for homogeneous agent living in a complete network to observe the effect of this kind world on the market dynamic. Finally, in section 6 we conclude and discuss the future extension of the research.

2 Literature Reviewer

The advancement of the agent-based computational economic model can be traced back to Frankel and Froot [11] in 1986. In 1987, Kim and Markowitz [18] built up a straightforward agent-based model to two sorts of investors to examine the fluctuation in the asset prices. Gode and Sunder [14] created a zero-intelligence based market under budget and no budget constraint and they infer that that the market converges to equilibrium with the inclusion of the budget constraint.

The development of the Santa Fe artificial market [2] propelled the utilization of the genetic algorithm to mimic the agent ability of adaptation. In addition, Brock and Hommes [5][6] created an agent-based model with switching mechanism between the trading strategies upon the agent's utility functions. Likewise, Chan et al.[7] developed an agent-based model that that consider informed traders, partially informed traders, and uninformed traders. Takahashi et al.[28] developed an agent-based model that incorporate loss aversion bias and fundamental and technical investment strategies.

The majority of the studies assumed that the agents interact indirectly through the posted asset prices. In any case, Panchenko et al. [27] expanded Brock and Hommes [5][6] by by looking at the impact of various types of network topologies on the dynamic of asset prices. The network topologies include complete network lattice network, small world network and random network. Nevertheless, in this paper, we rather concentrate on the effect of network strutter with the domain of scale-free network topology. For in-depth survey, scholar may explore [1][15][22]

3 Description of Meta-model for the Financial Market

The investigation of market dynamics in this paper was performed based on the model that developed by Khashanah and Alsulaiman [17]. In their model, Khashanah and Alsulaiman have divided the market into macro and micro levels.

The agents in the developed market live in an environment that is represented in term of scale-free network topology. They may interact through this network according to the parameter H where H represents the initial number of hubs in the network according to the preferential attachment algorithm [3].

In the macro level, there is one type of agents representing the market regulator. The market regulator may control the market through various tools such as risk-free rate, tax on transactions, restriction rules on holding and short selling of the equities.

In the micro level, there is one type of agents that represents various traders and investors in the stock market. These traders and investors are divided into 24 types distinguished in terms of their investment strategies and their investment behaviors. four investment strategies are involved in the developed market. These strategies represent the zero-intelligence investors, fundamental investors, momentum investors and adaptive investors. on the other hand, four behaviors are developed in the market that includes risk averse investors, loss averse investors, overconfidence investors and conservative investors.

Zero intelligence traders (also may be called noise traders) speculate on the stock price randomly on a range of specified prices. They may be viewed as the random walkers of the market. In contracts, the fundamental investors observe the prices around the fundamental value of the stock where the fundamental stock prices follow a geometric Brownian motion. The momentum traders follow the trend of the stock, i. e. if the price trends up in the last trading session they expect that it will continue to rise and vice versa. The adoptive investors utilize an artificial neural network to speculate on future prices using accumulated learning abilities. They adapt to the new conditions of the market by optimizing the weights in the neural network. In our model, we equip the optimization of the weights with probability parameter K indicating the propensity to change and adapt.

For tractability, the market is limited to trading one risk-free asset and one risky stock. Their holdings of the stock is constructed based on the following equation:

$$x_{i,j,t}^* = \frac{E_{i,j,t}(p_{t+1} + d_{t+1}) - (1 + r_f) p_t \pm c p_t}{\lambda_{i,j} v_{i,j} \beta_{i,j} \sigma_{i,t,p_{t+1}+d_{t+1}}^2}$$

where $E_{i,j,t}(p_{t+1} + d_{t+1})$ is the expected price and dividend for the next time step, which is crucial for the determination of the optimal holding. The expectations of heterogeneous agents are by necessity diverse and they are determined based on the investment strategies explained in the next section. Here $\sigma_{i,t,p_{t+1}+d_{t+1}}^2$ is the conditional standard deviation of price and dividend at time $t + 1$. For simplicity, $\sigma_{i,t,p_{t+1}+d_{t+1}}^2$ is assumed to be fixed and constant at a value of 1. The change in the sign in the above equation opposite the state of $E_{i,j,t}(p_{t+1} + d_{t+1}) - (1 + r_f) p_t$ makes $x_{i,j,t}^* = 0$. By the change in the sign, we mean that the negative sign follows the positive state of $E_{i,j,t}(p_{t+1} + d_{t+1}) - (1 + r_f) p_t$ and the positive sign follows the negative state of $E_{i,j,t}(p_{t+1} + d_{t+1}) - (1 + r_f) p_t$.

The traditional assumption in economic models is that agents are risk averse and, based on that assumption, economic models establish the objective function. Risk aversion implies that the agent would value certain outcomes over uncertain ones. The risk aversion coefficient λ_i may affect the agent decision to hold the stock.

The risk averse assumption has been opposed by the prospect theory. In prospect theory, Tversky and Kahneman [16] experimentally demonstrated that individuals have a bias to stress misfortunes more than benefits and consequently they are more loss disinclined than risk averse. The loss averse investors have different utility functions to a pre-determined reference point. In our model, the change of investor's wealth is set to be the reference point where the value of parameter β increases once the investor's wealth is exposed to a negative change.

Overconfidence investors tend to hold higher positions of the risky asset as they have more prominent trust in their decisions. On the other hand, conservative investors tend to hold lower positions in risky assets. The parameter v_i is less than one if the trader is overconfident and greater than one if the trader is conservative.

Agents may change their initial decisions of stock holding x_i^* as a consequence of the interaction with other agents. Whenever agents have a direct interaction with each other with a chance to share their sentiment on the market, agents may be

influenced to change their outlook on the market. The final holding decision X_i^* is then constructed as the weighted average of the agent initial decision and the initial decisions of the connected agents.

$$X_{i,j,t}^* = \begin{cases} \alpha_{i,j} x_{i,j,t}^* + \frac{(1+\alpha_{i,j})}{\sum_{j=1, j \neq i}^N I_k} x_{i,j,t}^* & \text{if } connections > 0 \\ x_{i,j,t}^* & \text{otherwise} \end{cases} \quad (1)$$

where $X_{i,j,t}^*$ is the final decision for agent i and α is a given weight for the initial decision of holding shares of stock $x_{i,j,t}^*$ for agent i . N is the total number of agents and I_k is:

$$I_k = \begin{cases} 1 & \text{if agent } k \text{ is connected to agent } i \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

In addition, agents may adapt different investment strategies and behaviors throughout the period of the simulation. They may imitate the investment strategy and behavior of the wealthier agent that is the subnetwork.

The price formation will follow the price adjustment method [4][26][8][23]. The price adjustment method set up the market price based on the aggregate bids and offers.

$$p_t = p_{t-1} [1 + \eta (B_t - O_t)] \quad (3)$$

Where p_t is the market price at time t , η is the price adjustment speed relative to the spread, i.e., a simplified form of market efficiency. Further, B_t represents the total number of bids among all agents and O_t is the total number of offers.

4 Properties of the Calibrated Network

Understanding the complex systems required an appropriate comprehension of its structured network. The complex system contains distinctive kind of agents that may communicate in different ways. These agents and their communication may be represented as nodes and edges in the network terminologies. We consider certain interactions and agents as a subsystem of the whole system. Consequently, the network structures of these agents and interactions can be displayed as subnetworks of the entire network (or what is called the network of networks). The networks of networks can be defined as multiple networks that are interconnected with each other [9][13][29]. Mikko Kivelä et al. [19] provided a comprehensive discussion, history, similitude and contrasts of different notations of complex networks. The discussion includes the terminologies and frameworks of multilayer networks, multiplex network, multivariate networks, networks of networks and many others.

The meta-model was fitted to the S&P500 from 2010 to 2014 by calibrating the various model parameters. These parameters include population size of agents

as they are classified by two-dimensional attributes according to their investment behaviors and investment strategies. Six investment behaviors are incorporated in the model which they are risk aversion (R), risk aversion with overconfidence and conservative biased (RO) and (RC), loss aversion (L) and loss aversion occupied with overconfidence and conservative biased (LO) and (LC). The investment strategies are zero-intelligence (Z), fundamental strategy (F), trend followers (T), artificial neural network (N). Table 1 shows the calibrated population size of the agents. The calibration was performed using scatter search metaheuristic algorithm through OptQuest machine [20][21].

Table 1: Calibrated population size of agents types

	Z	F	T	N
R	7	8	10	8
RO	10	7	10	10
RC	7	8	10	9
L	8	9	10	9
LO	7	7	7	10
LC	11	14	7	8

The average moments of stock returns of 1000 Monte Carlo simulation runs were 0.00052, 0.00912, 0.957 and 5.94 for the mean, standard deviation, skewness, kurtosis, respectively.

However, the stock prices are not driven by the population sizes only but rather by the structure of the network. In this paper, we examine the properties of the calibrated network. The calibrated network contains 211 nodes (agents) and 1157 edges (connections). The structure of the network is presented in figure 1 where the nodes represent the agents who participate in the market and the edges demonstrate the connection between them. The nodes were colored according to our classification of the two-dimensional subnetworks upon the behavior and strategy.

We define a subnetwork as the subset of agents (nodes) with similar two-dimensional attributes of behavior and strategy. Let a be an agent with attribute vector (b, s) with b, s referring to behavior and strategy, respectively. Let a_1, a_2 be two agents with coordinates $(b_1, s_1), (b_2, s_2)$ and define the relationship of $a_1 \sim a_2$ if and only if $b_1 = b_2$ and $s_1 = s_2$. Then clearly this relation creates equivalence classes and a partition of the set of agents (nodes). A subnetwork is defined as an equivalence class under the relation \sim .

The centrality of a node describes the importance of that node in the network. The centrality of nodes is computed using the normalized degree centrality, normalized closeness centrality, and normalized betweenness centrality.

There are 23 agents representing the 90 percentile of the most connected agents in terms of the degree measure and 22 agents in terms of closeness and betweenness. The majority of most connected agents are from risk averse fundamental agents

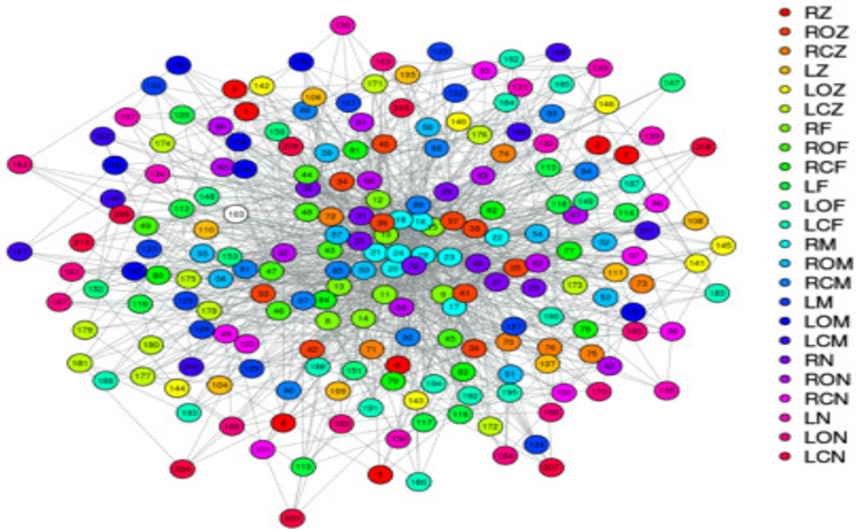


Fig. 1: Calibrate network

and risk averse momentum agents types where the agents from RF1 to RF8 are characterized among the most connected agents in all of the three measures. Also, the agents from RM1 to RM10 are classified as the most connected agents for all of the centrality measures except agent RM7 for the betweenness measure. In addition, agent RN1, RN2, RN7, and RN8 are from the 90 percentile of the most connected agents for all of the three centrality measures except for RN7 for closeness centrality. Agent RON1 is among the 90 percentile of the most connected agents for the degree and betweenness measures and agent ROZ5 is in the top of the most connected agents according to closeness centrality measure.

We need a network measure for expressing the proverb “birds of feather flock together” and the contrary saying “opposites attract”. The first idea is measured by assortativity and the second one is measured by disassortativity. The degree assortativity coefficient (ac) measures the level of the homophily of the network, for example, the hubs have the tendency to connect with the hubs [24][25]. The value of ac in the calibrated network, was -0.0839237 which indicates that the network is non-assortative. We have computed the assortativity of the network in terms of degree connectivity (DC), two-dimensional attributes based on behavior and strategy (TD), one-dimensional attribute based on behavior (B) and one-dimensional attribute based on strategy (S). The results were -0.0839237 , -0.02338304 , 0.004135164 and -0.02775979 for DC, TD, B, and S respectively which indicates that the network is non-assortive for all criteria.

In addition, the centralization score is measured in terms of the degree, betweenness, closeness, and eigenvector. The centralization is a measure of network centrality that depends on the centrality of the nodes. The most centralized network in terms

of degree, betweenness and closeness is a network that resembles a star-network. However, the most centralized eigenvector network is a network with few edges that connect few nodes [12] [30]. The centralization scores for the calibrated network were 0.195, 0.0785, 0.284 and 0.785 in term of degree, betweenness, closeness, and eigenvector respectively which indicate a low level of centrality in terms of degree, betweenness and closeness and a high level of centrality in term eigenvector.

Furthermore, we investigated the cliques in the network. The clique is a subset of the network where all the nodes are connected to each other. The maximal clique is the clique that can't be extended to a larger clique and the maximum clique is the largest clique in the network[10]. The size of the largest cliques in the network was 6. These largest cliques are (RF5, RF7, RM1, RM2, RM3 and RM7), (RF5, RF7, RF3, RM1, RM3 and RM7), (RM7, RM3, RF5, RF7, RF1 and RM2), (RF1, RF5, RF6, RM3, RM2 and RM7), (RF1, RF5, RM7, RF7, RM0 and RM2), (RF5, RF7, RM2, RM0, RM4 and RM6), (RF2, RF5, RF7, RM1, RM2 and RM4), (RF2, RF5, RF7, RM0, RM2 and RM4), (RF0, RF5, RF7, RM0, RM4 and RM6), (RF2, RF5, RF7, RM1, RM2 and RM3) and (RF1, RF5, RF7, RM0, RM2 and RM6).

5 Market Emergence Under Homogeneous Complete Network

In this section, we observe the patterns of stock prices, returns, market capitalization and wealth distributions under a homogeneous environment with a complete network structure. Four experiments were implemented for this purpose by which each experiment contains 211 agents that all are risk averse. In the first experiment, agents are zero-intelligence while agents in the second and third experiments utilized fundamental and trend following investment strategies, respectively. In the fourth experiment, agents are adaptive agents who use ANN.

The mean of the daily returns, standard deviation, skewness, kurtosis and the market capital at time $T = 1000$ are shown in table 2. In addition, the patterns of the prices, returns and market capital for all experiments are shown in Appendix A. These results are significantly different than the results of calibrated heterogeneous market which indicate that both the network structure and agents heterogeneity significantly impact the emergence of patterns in the financial market. The market with zero-intelligence agents tends to have higher volatility and lower kurtosis than the calibrated market. On the other hand, the market volatility is much less than the calibrated market when it is bind with fundamental, trend followers or adaptive agents.

In terms of wealth distributions, the average wealth of the 211 agents at $T = 1000$ in the first experiment was 110,197.83 and the standard deviation was 1142.8. In the second experiment the average wealth was 111,241.42 and the standard deviation 328.82 and in the third experiment, the average wealth was 110,516.54 and the standard deviation zero because the no transactions were ever executed. The average of wealth in the fourth experiment was 110,516.9 and the standard deviation is 3.97. These results in comparison to the calibrated heterogeneous market results show that the wealth distribution among agents is less concentrated when markets contain

Table 2: Statistics of the emergence market patterns

	Zero-intelligence	Fundamental	Trend followers	Adapitaive
Mean return s	-1.65E-05	1.27E-04	-2.13E-04	-1.10E-04
Volatility	0.013	0.003	0	8.41E-05
Skwness	0.069	0.743	-0.192	3.542
Kurtosis	0.138	3.73	-0.094	16.569
Market capital	23,251,741.45	23,471,940.22	23,318,989.73	23,316,658.12

more homogeneous agents. Figure 2 shows the wealth distributions for the four experiments.

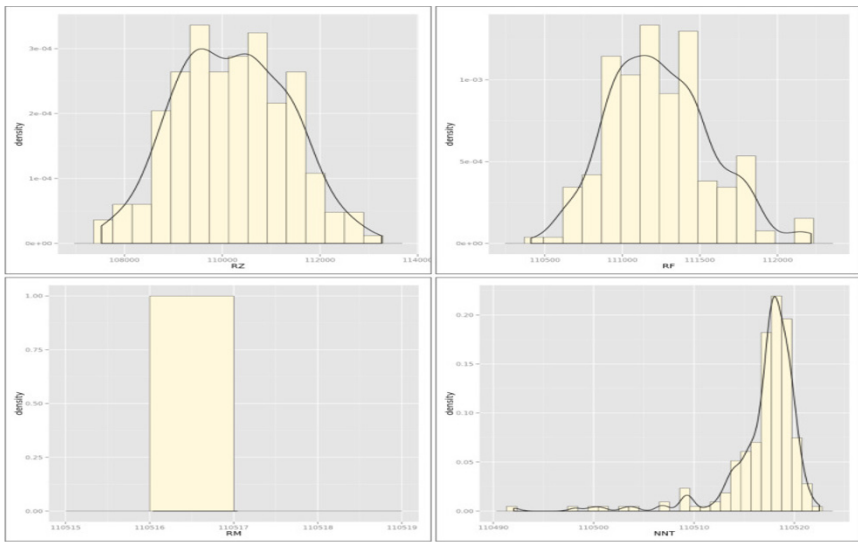


Fig. 2: Wealth distributions

6 Conclusion and Future Research

The paper is designed to examine the effect of network structure and agents attributes on the emergent behaviors of a simulated financial market. Two main objectives were in the aim of the study. The first objective is to introduce the concept of a network of networks for simulated markets. The goal was accomplished by investigating the properties of a calibrated network of the financial market of a pre-developed agent-based model and compare these properties to a simplified network of multiple subnetworks.

The second objective is to investigate the evolution of stock prices, returns, market capital and wealth distribution under a population of a homogeneous agent that are connected in a complete network structure. Four experiments were implemented. In all experiments, we assumed that the agents are risk averse. In the first experiment, all agents are zero-intelligence while in the second experiment agents are fundamental. In the third the fourth experiments, agents are adaptive. In general, the moments of the market in all of the were significantly different than the calibrated moments, especially in the standard deviation and kurtosis. However, the standard deviation is higher when the market is occupied with zero-intelligence agents and lower with fundamental agents trend followers agents, and adaptive agents. The kurtosis was lower than the calibrated kurtosis in all four cases. However, the market capital does not exhibit a significant difference from the calibrated market. Furthermore, the variation of wealth when the agents are homogeneous are much less than the variation of heterogeneous market environment.

In an extension of this research, we would examine the market under different network structures. Also, we would elaborate more on the concept of a network of networks by implementing multiple comparisons of newly developed networks and their aggregations.

References

- [1] Alsulaiman, T., Khashanah, K.: Bounded rational heterogeneous agents in artificial stock markets: Literature review and research direction. *International Journal of Social, Behavioral, Educational, Economic and Management Engineering* **9**, 2038–2057 (2015)
- [2] Arthur, W.B., Holland, J.H., LeBaron, B., Palmer, R.G., Tayler, P.: Asset pricing under endogenous expectations in an artificial stock market. Available at SSRN 2252 (1996)
- [3] Barabási, B.A.L., Bonabeau, E.: Scale-free. *Scientific American* (2003)
- [4] Bertella, M.A., Pires, F.R., Feng, L., Stanley, H.E.: Confidence and the stock market: An agent-based approach. *PloS one* **9**(1), e83,488 (2014)
- [5] Brock, W.A., Hommes, C.H.: A rational route to randomness. *Econometrica: Journal of the Econometric Society* pp. 1059–1095 (1997)
- [6] Brock, W.A., Hommes, C.H.: Heterogeneous beliefs and routes to chaos in a simple asset pricing model. *Journal of Economic dynamics and Control* **22**(8), 1235–1274 (1998)
- [7] Chan, N.T., LeBaron, B., Lo, A.W., Poggio, T., Yy, A.W.L., Zz, T.P.: Agent-based models of financial markets: A comparison with experimental markets. Citeseer (1999)
- [8] Derveeuw, J.: Market dynamics and agents behaviors: a computational approach. In: *Artificial Economics*, pp. 15–26. Springer (2006)
- [9] Donges, J.F., Schultz, H.C., Marwan, N., Zou, Y., Kurths, J.: Investigating the topology of interacting networks. *The European Physical Journal B* **84**(4), 635–651 (2011)
- [10] Eppstein, D., Löffler, M., Strash, D.: Listing all maximal cliques in sparse graphs in near-optimal time. In: *International Symposium on Algorithms and Computation*, pp. 403–414. Springer (2010)
- [11] Frankel, J.A., Froot, K.A.: Explaining the demand for dollars: International rates of return and the expectations of chartists and fundamentalists. Department of Economics, UCB (1986)
- [12] Freeman, L.C.: Centrality in social networks conceptual clarification. *Social networks* **1**(3), 215–239 (1978)
- [13] Gao, J., Buldyrev, S.V., Havlin, S., Stanley, H.E.: Robustness of a network of networks. *Physical Review Letters* **107**(19), 195,701 (2011)

- [14] Gode, D.K., Sunder, S.: Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of political economy* pp. 119–137 (1993)
- [15] Hommes, C.H.: Heterogeneous agent models in economics and finance. *Handbook of computational economics* **2**, 1109–1186 (2006)
- [16] Kahneman, D., Tversky, A.: Prospect theory: An analysis of decision under risk. *Econometrica: Journal of the econometric society* pp. 263–291 (1979)
- [17] Khashanah, K., Alsulaiman, T.: Network theory and behavioral finance in a heterogeneous market environment. *Complexity* (2016)
- [18] Kim, G.r., Markowitz, H.M.: Investment rules, margin, and market volatility. *The Journal of Portfolio Management* **16**(1), 45–52 (1989)
- [19] Kivela, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., Porter, M.A.: Multilayer networks. *Journal of complex networks* **2**(3), 203–271 (2014)
- [20] Laguna, M., Marti, R.: The optquest callable library. In: *Optimization Software Class Libraries*, pp. 193–218. Springer (2003)
- [21] Laguna, M., Marti, R.: *Scatter search: methodology and implementations in C*, vol. 24. Springer Science & Business Media (2012)
- [22] LeBaron, B.: Agent-based computational finance. *Handbook of computational economics* **2**, 1187–1233 (2006)
- [23] Martinez-Jaramillo, S., Tsang, E.P.: An heterogeneous, endogenous and coevolutionary gp-based financial market. *IEEE Transactions on Evolutionary Computation* **13**(1), 33–55 (2009)
- [24] Newman, M.E.: Assortative mixing in networks. *Physical review letters* **89**(20), 208,701 (2002)
- [25] Newman, M.E.: Mixing patterns in networks. *Physical Review E* **67**(2), 026,126 (2003)
- [26] Palmer, R.G., Arthur, W.B., Holland, J.H., LeBaron, B., Tayler, P.: Artificial economic life: a simple model of a stockmarket. *Physica D: Nonlinear Phenomena* **75**(1), 264–274 (1994)
- [27] Panchenko, V., Gerasymchuk, S., Pavlov, O.V.: Asset price dynamics with heterogeneous beliefs and local network interactions. *Journal of Economic Dynamics and Control* **37**(12), 2623–2642 (2013)
- [28] Takahashi, H., Terano, T.: Agent-based approach to investors' behavior and asset price fluctuation in financial markets. *Journal of artificial societies and social simulation* **6**(3) (2003)
- [29] Wang, Y., Xiao, G.: Effects of interconnections on epidemics in network of networks. In: *Wireless Communications, Networking and Mobile Computing (WiCOM)*, 2011 7th International Conference on, pp. 1–4. IEEE (2011)
- [30] Wasserman, S., Faust, K.: *Social network analysis: Methods and applications*, vol. 8. Cambridge university press (1994)

7 Appendix A

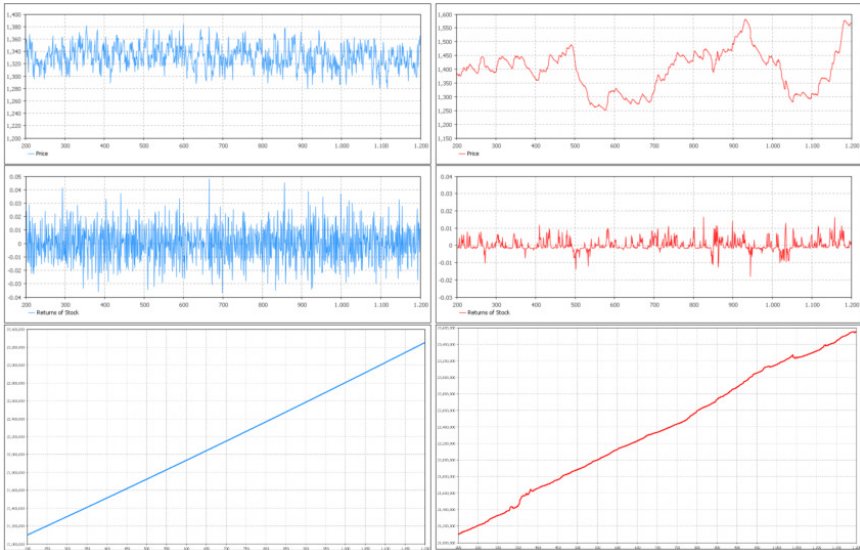


Fig. 3: Market emergence behaviors (zero-intelligence agents(left) and fundamental agents (right))

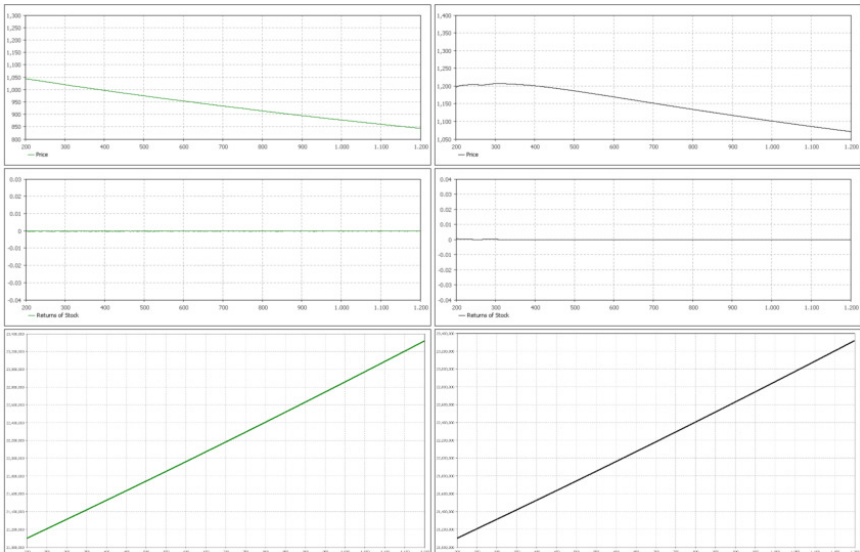


Fig. 4: Market emergence behaviors (trend followers (left) and adaptive agents (right))

Part X
Biological and Ecological Networks

Motif-Based Analysis of Effective Connectivity in Brain Networks

J. Meier, M. Märtens, A. Hillebrand, P. Tewarie and P. Van Mieghem

Abstract Network science has widely studied the properties of brain networks. Recent work has observed a global back-to-front pattern of information flow for higher frequency bands in magnetoencephalography data. However, the effective connectivity at a local level remains yet to be analyzed. On a local level, the building blocks of all networks are motifs. In this study, we exploit the measure of dPTE to analyze motifs of the estimated effective connectivity networks. We find that some 3- and 4-motifs, the bidirectional two-hop path and its extended 4-node versions, are significantly overexpressed in the analyzed networks in comparison with random networks. With a recently developed motif-based clustering algorithm we separate the effective connectivity network in two main clusters which reveal its higher-order organization with a strong information flow between posterior hubs and anterior regions.

J. Meier (e-mail: j.m.meier@tudelft.nl) · M. Märtens (e-mail: m.maertens@tudelft.nl) · P. Van Mieghem (e-mail: p.f.a.vanmieghem@tudelft.nl)
Delft University of Technology, Faculty of Electrical Engineering, Mathematics and Computer Science, P.O. Box 5031, 2600 GA Delft, The Netherlands

A. Hillebrand (e-mail: a.hillebrand@vumc.nl)
Department of Clinical Neurophysiology and Magnetoencephalography Center, Neuroscience Campus Amsterdam, VU University Medical Center, Amsterdam, The Netherlands

P. Tewarie (e-mail: prejaas.tewarie@nottingham.ac.uk)
Department of Neurology, VU University Medical Center, Amsterdam, The Netherlands.
Sir Peter Mansfield Imaging Centre, School of Physics and Astronomy, University of Nottingham, University Park, Nottingham, The United Kingdom

1 Introduction

Analyzing the brain as a network has led to new insights in neuroscience both in understanding healthy and abnormal brain function [22]. Recent studies in neuroscience applied the measure of Phase Transfer Entropy (PTE) to construct the effective connectivity network between brain regions and observed a global posterior-anterior pattern in higher frequency bands [10]. However, the effective connectivity at a local level remains yet to be analyzed. In this study, we analyze with PTE the directionality at a local level in the form of network motifs.

Effective connectivity describes the causal effect of one brain region on another region [1, 7]. To calculate this pairwise value between brain regions, the measure of Transfer Entropy (TE) is often applied [19]. The TE from a region X to a region Y quantifies the improvement in predicting the future of time series X if the present value of Y is also included. Recent work has extended this measure to the analysis of phase time series (Phase Transfer Entropy (PTE); [15]). The advantage of phase time series instead of the original time series is the lower computational cost for analyzing their pairwise interactions [18]. When representing brain regions as nodes and assigning PTE values as link weights, one can build the effective connectivity network. A previous study used PTE for magnetoencephalography (MEG) data from healthy controls and discovered a posterior-anterior directionality in the effective connectivity network of all frequency bands except for the *theta* band (where the pattern was opposite) [10]. The emergence of this pattern is still not completely understood. The hypothesis was that this global directionality is caused by different local properties in the effective connectivity network [10].

On a local scale, network motifs are the building blocks of all networks [17]. On top of the micro-structure of nodes and links, network motifs are small subgraphs that form a higher-order organization of the network [4]. Most commonly, network motifs of 3 or 4 nodes are analyzed. Friedman et al. [6] were recently able to identify Alzheimer patients with directed motif analysis in a so-called progression network. Previous work reported that the motif with ID 78 was overexpressed with respect to random networks in the structural brain networks of the cat and the macaque [21] (see Fig. 2 for motif IDs). The same motif has also been perceived as a good identifier for structural hubs [11]. Recently, Battiston et al. analyzed the interdependency between structure and function in the human brain applying a multilayer motif approach [3]. With computational models of neuronal activity, Battaglia and co-authors [2] linked effective connectivity motifs based on TE to underlying structural motifs and suggested that changes in the effective connectivity lead to different global directions of information flow. With similar motivation of linking frequencies of single motifs to global outcomes, Benson et al. [4] exploited this higher-order organization of the network to define a new motif-based clustering algorithm.

The aim of this study is to investigate effective connectivity motifs in empirical data with the measure of PTE. Therefore, we first explain the construction of the effective connectivity network based on the sending and receiving properties of a node. Then, we analyze the significant motifs in this network. Furthermore, we apply the recently developed motif-based clustering algorithm by Benson et al. [4] on the effective connectivity brain network.

2 Methods

This section explains the measure of directed Phase Transfer Entropy (dPTE), the construction of the directed networks, the motif search and our application of the motif-based clustering.

2.1 Directed Phase Transfer Entropy

The effective connectivity network is based on MEG measurements¹ of 67 healthy controls from a preceding study [10]. We focus our analysis on the *alpha2* frequency band (10-13 Hz) because the previous study observed a significant pattern of posterior-anterior information flow for this frequency band. For every region of interest (ROI) X we compute a time series in the form of a phase time series [18]. We denote a possible value of the signal of region X at time t by x_t and abbreviate the probability that the signal of X equals x_t at an arbitrary time point t to $\Pr[X_t = x_t] = \Pr[x_t]$. The information flow between two ROIs or nodes, X and Y , is then quantified by the Phase Transfer Entropy [15]

$$PTE_{XY}(h) = \sum \Pr[x_{t+h}, x_t, y_t] \times \log \left(\frac{\Pr[x_{t+h}|x_t, y_t]}{\Pr[x_{t+h}|x_t]} \right), \quad (1)$$

for a certain time delay h , where the sum runs over all possible values x_t , x_{t+h} and y_t of the signals. The (joint) probabilities are determined over histograms of their occurrences in an epoch [15]. Following Hillebrand et al. [10] we fix h at

$$h = \frac{N_s \cdot N_{ROI}}{N_{\pm}}, \quad (2)$$

where $N_s = 4096$ and $N_{ROI} = 78$ are the number of samples and the number of ROIs, respectively, and N_{\pm} counts the number of sign changes for the phase across time and ROIs.

¹ The MEG data were recorded using a 306-channel whole-head MEG system (ElektaNeuromag, Oy, Helsinki, Finland) during a no-task, eyes-closed condition for five consecutive minutes. A beamformer approach was adopted to project MEG data from sensor space to source space [9] and the automated anatomical labelling (AAL) atlas was applied to obtain time series for 78 cortical regions of interest (ROIs) [8, 24]. For each subject, we extracted the first 20 artefact-free epochs of 4096 samples (3.2768 s).

Motivated by Hillebrand et al. [10], we define the dPTE for nodes X and Y as

$$dPTE_{XY} = \frac{PTE_{XY}}{PTE_{XY} + PTE_{YX}}, \tag{3}$$

which is a measure of the preferred direction of information flow between nodes X and Y . Since the PTE can only take positive values, this definition of dPTE is well-defined and its value ranges from 0 and 1. If the predominant flow of information is from node X to node Y , then $0.5 < dPTE_{XY} < 1$, else $0 < dPTE_{XY} < 0.5$.

2.2 Constructing the Directed Network

The pairwise dPTEs over all ROIs can be interpreted as a weight matrix of a fully connected network. Since the data is from 67 subjects each over $k = 20$ epochs, we have 1340 weighted networks to begin our construction. We apply a procedure to thin out links and induce a directionality per link instead of a weight. After this transformation, which we call “sparsification”, we obtain a sparse directed (unweighted) network for each subject, which is amenable for motif search and analysis.

The sparsification (see Fig. 1) contains two steps. First, we discard all links whose weights are in close proximity to 0.5. More precisely, every link whose average weight (over all epochs) is within the closed interval $[0.5 - \alpha\sigma, 0.5 + \alpha\sigma]$ will not be considered, where σ is the standard sample deviation taken over all epochs over all pairs of nodes and α is a positive real control parameter. Under the assumption of a normal distribution with mean 0.5, the 3σ -rule states that this procedure will remove approximately 68% for $\alpha = 1.0$ and 95% for $\alpha = 2.0$ of all links.

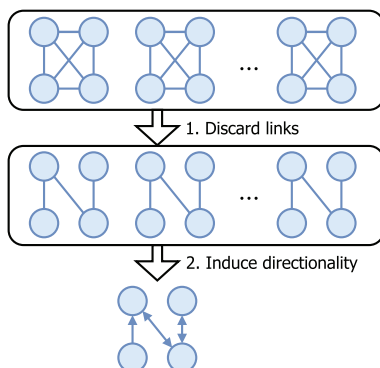


Fig. 1 Schematic overview of the two steps for constructing the directed network (sparsification): (1) discard links close to 0.5 (2) induce directionality for remaining links.

In a second step, we determine for each remaining link whether it should be bi- or uni-directional, and in case of the latter, in which direction the links should be oriented. Clearly, all remaining link weights are now bounded away from 0.5, though it is possible, that for different epochs a single link weight might be lower or higher than 0.5, which makes it ambiguous which member of the node pair is the dominant sender and which the dominant receiver. Let k^+ (k^-) be the number of epochs that the $dPTE_{XY}$ is above (below) 0.5 where $k = k^+ + k^-$ is the total number of epochs for a subject. If $k^+/k \geq 0.75$, we assume X to be a dominant sender and thus we induce a uni-directional link from X to Y . Contrary, we assume X to be a dominant receiver if $k^+/k \leq 0.25$ and point the link from Y to X . If neither applies ($0.25 < k^+/k < 0.75$), we assume that X and Y frequently change roles between dominant sender and dominant receiver. Thus, we induce a bidirectional link between them.

2.3 Motif Search

We are using the excellent *mfinder* software [13], provided by the Uri Alon Lab², to search for motifs. We also adopted the motif IDs of *mfinder* for this work, to be consistent. With sparsification, we generate one directed network for each of the 67 subjects as input for *mfinder*. Additionally, we construct an averaged effective connectivity network (short: averaged network) by considering all epochs of all subjects together. This construction results in a “virtual” subject with $k = 1340$ instead of $k = 20$ epochs. We set α to 1.0 and 2.0 to compare on different levels of sparsity.

Since the complexity of motif search increases dramatically with the size of the motif, we restrict *mfinder* to search only for subgraphs of 3 and 4 nodes (further called 3-motifs and 4-motifs). The *mfinder* program executes two tasks: first, it counts the frequency of all motifs in the original input network. Second, it generates a number of random networks (null model) and determines the motif frequencies in each of them as well. In total, *mfinder* generates 1000 random networks using the switching algorithm described by Milo et al. [16] for each single input network. We use the default parameters for *mfinder*, which preserve the degree sequence of the original network and the number of bidirectional links.

A motif is called overexpressed if it occurs significantly more often in the original network than in the random networks. It is essential to keep in mind that a motif which is not overexpressed may still occur quite frequently in the original network, though it arises in a similar frequency by a random link rewiring process. Thus, it can be argued that overexpressed motifs must carry some functional importance for the underlying system since they do not arise merely by chance. We report the motifs that *mfinder* determines to be overexpressed with z -score > 2 .

² <https://www.weizmann.ac.il/mcb/UriAlon/download/network-motif-software>

2.4 Motif-Based Clustering Algorithm

Benson et al. [4] developed a clustering algorithm that partitions a network based on one specific overexpressed motif M . The algorithm constructs clusters by 'cutting' through the minimal possible number of those motifs. Formally, the clustering minimizes the motif conductance defined as

$$\phi_M(S) = \frac{\text{cut}_M(S, S^c)}{\min[\text{vol}_M(S), \text{vol}_M(S^c)]}, \quad (4)$$

where S is the set of nodes in the cluster and S^c its complement. Here, $\text{cut}_M(S, S^c)$ is the number of M motifs that is cut through and $\text{vol}_M(S)$ the number of M motifs that is completely in S . The algorithm can be regarded as an extension of the classic spectral clustering algorithm [25]. The obtained clusters reveal a higher-order organization of the network based on the specific motif M . An implementation of the motif-based clustering algorithm was released as part of the open SNAP framework [14], which we applied to the averaged network using default parameters.

3 Results

We present results for the motif search on 3 and 4 nodes for the individual subjects and for the averaged network, respectively. In addition, we show the results of the motif-based clustering algorithm on the averaged network.

3.1 Significant 3-Motifs

For both variants of the sparsification method ($\alpha = 1$ and $\alpha = 2$), we find the same significant 3-motifs over all subjects meaning that those motifs are more frequent in our analyzed networks than in the null model (see Fig. 2). Those five motifs are not triangular but include all 3-motifs with two links (except for the 2-hop path motif (Fig. 2b- 2f)). The absolute frequency of those motifs is displayed as a histogram in Fig. 2a for the $\pm\sigma$ and the $\pm 2\sigma$ sparsification, respectively. The analysis on the averaged effective connectivity network confirms the over-representation of the motif with ID 78, the bidirectional 2-hop path (Fig. 2d), which is the only significant motif that has been found for different sparsification methods (z-scores: 88.25 for $\pm\sigma$ sparsification and 82.7 for $\pm 2\sigma$ sparsification).

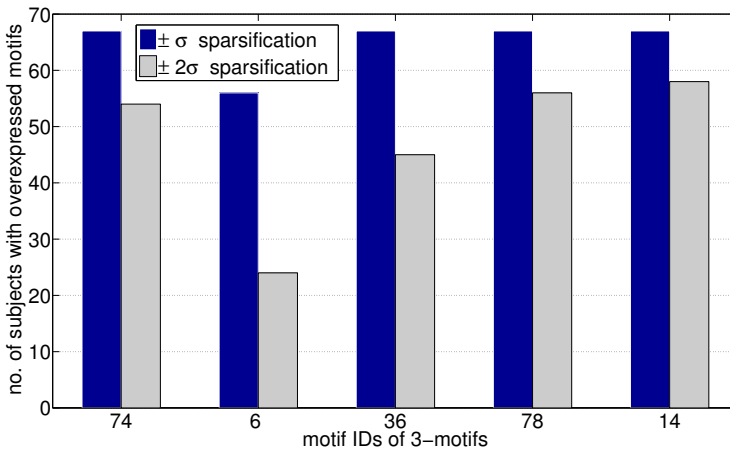
3.2 Significant 4-Motifs

In Fig. 3a we present a histogram of all significantly overexpressed 4-motifs with the two different sparsification levels. Twelve 4-motifs were found overexpressed in all 67 subject networks (Fig. 3a, for a visualization see Figs. 3b-3m).

Analyzing the averaged network we find 3 significant motifs with the $\pm\sigma$ sparsification method (see Figs. 3l - 3n, z-scores: 203.74 for ID 13260, 111.89 for ID 4382 and 14.85 for ID 4698) and none with the $\pm 2\sigma$ method. The two 4-motifs with number 13260 and 4382, the bidirectional ring and the bidirectional star, respectively, have the highest z-scores in the averaged effective connectivity network and are a subset of the significant 4-motifs found for every individual subject (Figs. 3l and 3m). The overexpression of those two motifs cannot be explained by the higher number of bidirectional links in the effective connectivity network since the null model contains the same number of bidirectional links.

3.3 Motif-Based Clusters

Following the approach of [4], we apply the motif-based clustering algorithm on the averaged effective connectivity network. Since for both sparsification methods, the 3-motif with ID 78 was significantly overexpressed in the averaged effective connectivity network and in every subject network, we cluster according to this motif. We find two clusters with the sparsified network for $\pm\sigma$ (Fig. 4). The frontal brain



(a) Histogram of all significantly overexpressed 3-motifs.

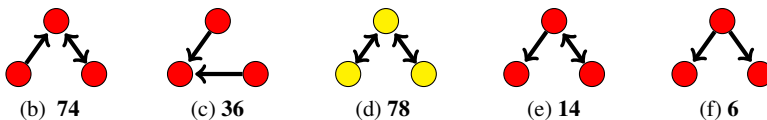
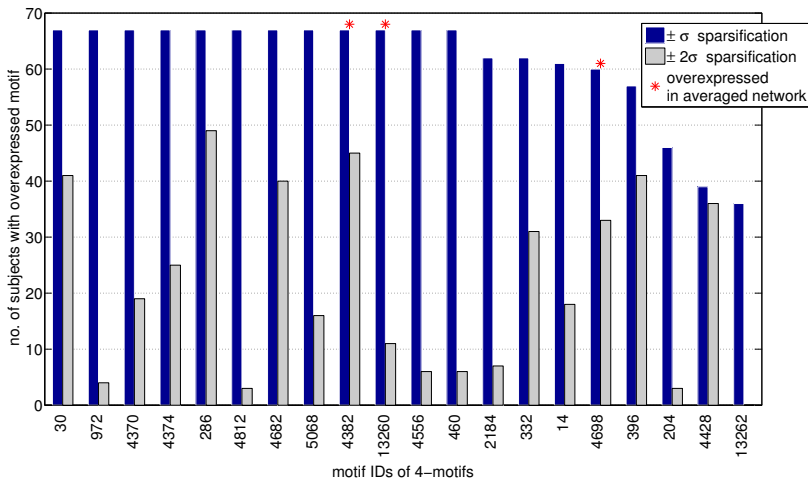


Fig. 2: (a) Frequency of significantly overexpressed 3-motifs over all regarded subjects after the $\pm\sigma$ and $\pm 2\sigma$ sparsification, respectively. (b)-(f) All significant 3-motifs over all subjects together with their motif ID. The yellow motif with ID 78 is also overexpressed in the averaged network.



(a) Histogram of the 20 most commonly overexpressed 4-motifs.

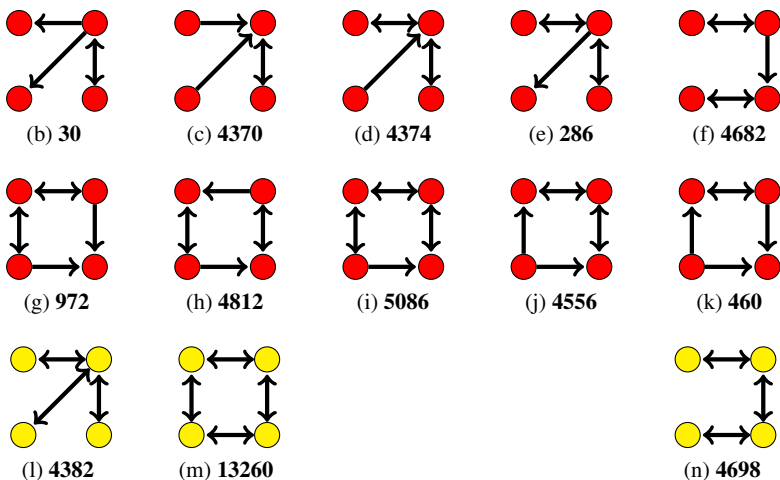
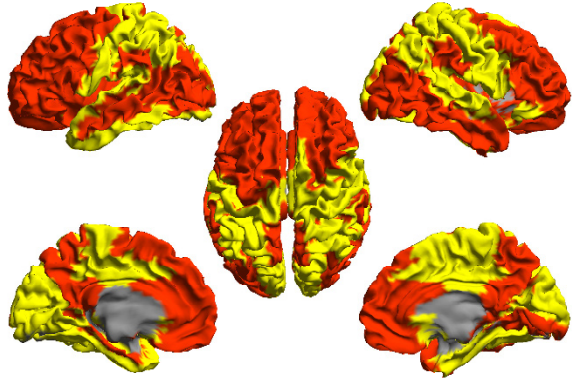


Fig. 3: (a) Histogram of the 20 most commonly overexpressed 4-motifs over all subjects after the $\pm\sigma$ and $\pm 2\sigma$ sparsification, respectively. An asterisk marks the motifs that are also overexpressed in the averaged network. (b)-(m) The twelve 4-motifs that are overexpressed after the $\pm\sigma$ sparsification in every subject with their motif ID. The yellow motifs are also overexpressed in the averaged network. (n) Third overexpressed 4-motif in the averaged network, ID 4698.

regions seem to be consistently part of the red cluster and the distribution of the clusters across the two brain hemispheres shows a strong symmetry (Fig. 4). The sparser network resulting from the $\pm 2\sigma$ sparsification method was disconnected.

Fig. 4 The two clusters (in red and yellow) on the template brain obtained via the motif-based clustering algorithm after the $\pm\sigma$ sparsification based on the motif 78.



Consequently, we could only obtain a motif-based clustering of the largest connected component (see Appendix Fig. 5).

4 Discussion and Conclusions

Evaluating the overexpressed motifs for individual human subjects, it is interesting that the 3-motif with ID 78 and its extended 4-node versions have also been overexpressed in other cortical networks of the cat and the macaque brain [21]. In these motifs some nodes seem highly integrated with their neighbors while others are more segregated. Sporns et al. [21] associated these motifs and the absence of triangular shapes with the general principles of integration and segregation in the functional organization of brain networks. This principle originates from studies of neuronal dynamics where signals from many different spatially segregated groups of neurons are integrated with each other forming one coherent signal [20, 23, 26]. In addition, motif 78 can help to identify hubs in structural brain networks by counting the number of times a node participates in that motif [11]. A possible explanation for this identification is that a hub often connects two otherwise disconnected brain regions reciprocally with each other functioning as a 'bridge' for the information flow [11]. Thus, the pre-dominance of motif 78 in the analyzed effective connectivity network suggests that hubs are 'bridges' for the information flow. The impact on the global network could be further investigated by the new metric of 'bridgeness' [12] in future research. Also the other significant 3-motifs are present in brain networks from the literature. For example, motif 6 has been identified in a previous modeling study with Granger causality as the driving structure behind many neuronal dynamics [5].

The fact that the motif-based clustering reveals a strong symmetry between the brain hemispheres is remarkable and supports the idea of a higher-order organization of the effective connectivity brain network. In comparison, the results of a standard spectral clustering algorithm (edge-based conductance) show a much weaker symmetry and a more disconnected spatial distribution of the two clusters (see Appendix Fig. 6). However, a rather dense network ($\pm\sigma$) seems to be necessary for the emergence of a higher-order structure since the clustering for the sparser averaged

network ($\pm 2\sigma$) appears to be frail (see Appendix Fig. 5). Thus, finding an optimal link density for motif-based clustering requires further investigation.

Looking into the obtained clusters, we find that the red cluster in Fig. 4 consists of all frontal brain regions and some posterior regions which are known to be the strongest structural hubs [10]. The fact that the motif-based clustering algorithm does not separate posterior hubs and frontal regions suggests that there might be an increased information flow between them. This result strengthens the hypothesis from [10] that the posterior hubs play a crucial role in the global information flow of the effective connectivity. More specifically, posterior hubs in the brain seem to play the role of a 'bridge' for not only the local but also the global information flow. However, this 'bridge' seems to be active in varying pre-dominant directions for different frequency bands [10]. To conclude, our study shows a promising way of integrating local structures to explain the emergence of global patterns in brain networks. This approach might be a first stepping stone towards understanding the information flow in the healthy brain which could, in the future, support the diagnosis of brain disorders.

Acknowledgements This work was partially supported by a private sponsorship to the VUmc MS center Amsterdam. The VUmc MS center Amsterdam is sponsored through a program grant by the Dutch MS Research Foundation (Grant number 09-358d). We thank Cornelis J. Stam for his useful comments and input that improved the paper. We are grateful to Jure Leskovec, who made his code for the motif-based clustering publicly available as part of the SNAP framework.

Appendix

Fig. 5 The two main clusters (in red and yellow) of the largest connected component on the template brain obtained via the motif-based clustering algorithm after the $\pm 2\sigma$ sparsification based on the motif 78. The blue colored regions were not in the largest connected component.

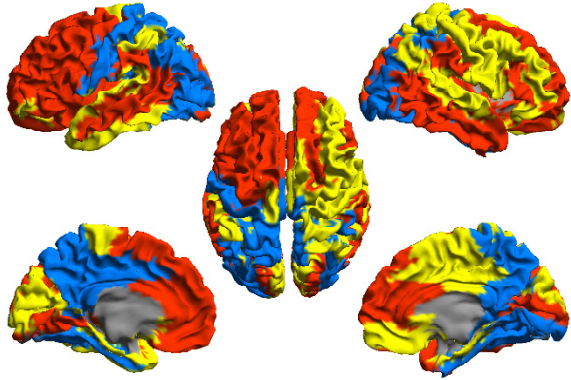
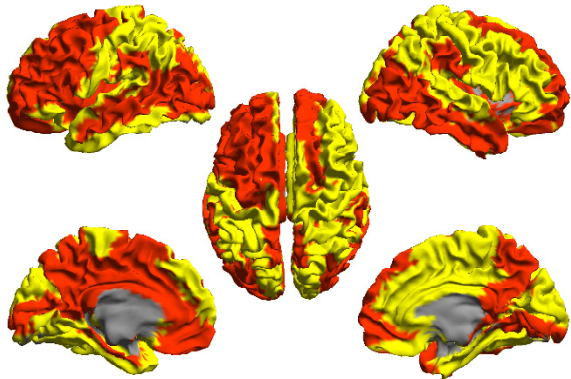


Fig. 6 The two main clusters (in red and yellow) on the template brain obtained via the spectral clustering algorithm with the $\pm\sigma$ sparsification. In comparison with the motif-based clustering in Fig. 4, the red cluster looks more disconnected and does not include all anterior regions anymore.



References

- [1] Aertsen, A., Gerstein, G., Habib, M., Palm, G.: Dynamics of neuronal firing correlation: modulation of "effective connectivity". *Journal of Neurophysiology* **61**(5), 900–917 (1989)
- [2] Battaglia, D., Witt, A., Wolf, F., Geisel, T.: Dynamic effective connectivity of inter-areal brain circuits. *PLoS Comput Biol* **8**(3), e1002438 (2012)
- [3] Battiston, F., Nicosia, V., Chavez, M., Latora, V.: Multilayer motif analysis of brain networks. arXiv preprint arXiv:1606.09115 (2016)
- [4] Benson, A.R., Gleich, D.F., Leskovec, J.: Higher-order organization of complex networks. *Science* **353**(6295), 163–166 (2016)
- [5] Deng, B., Deng, Y., Yu, H., Guo, X., Wang, J.: Dependence of inter-neuronal effective connectivity on synchrony dynamics in neuronal network motifs. *Chaos, Solitons & Fractals* **82**, 48–59 (2016)
- [6] Friedman, E.J., Young, K., Tremper, G., Liang, J., Landsberg, A.S., Schuff, N., Initiative, A.D.N., et al.: Directed network motifs in alzheimer's disease and mild cognitive impairment. *PLoS One* **10**(4), e0124453 (2015)

- [7] Friston, K.J.: Functional and effective connectivity in neuroimaging: a synthesis. *Human Brain Mapping* **2**(1-2), 56–78 (1994)
- [8] Gong, G., He, Y., Concha, L., Lebel, C., Gross, D.W., Evans, A.C., Beaulieu, C.: Mapping anatomical connectivity patterns of human cerebral cortex using in vivo diffusion tensor imaging tractography. *Cerebral Cortex* **19**(3), 524–536 (2009)
- [9] Hillebrand, A., Barnes, G.R., Bosboom, J.L., Berendse, H.W., Stam, C.J.: Frequency-dependent functional connectivity within resting-state networks: an atlas-based meg beam-former solution. *NeuroImage* **59**(4), 3909–3921 (2012)
- [10] Hillebrand, A., Tewarie, P., van Dellen, E., Yu, M., Carbo, E.W., Douw, L., Gouw, A.A., van Straaten, E.C., Stam, C.J.: Direction of information flow in large-scale resting-state networks is frequency-dependent. *Proceedings of the National Academy of Sciences* **113**(14), 3867–3872 (2016)
- [11] Honey, C.J., Kötter, R., Breakspear, M., Sporns, O.: Network structure of cerebral cortex shapes functional connectivity on multiple time scales. *Proceedings of the National Academy of Sciences* **104**(24), 10,240–10,245 (2007)
- [12] Jensen, P., Morini, M., Marton, K., Venturini, T., Vespignani, A., Jacomy, M., Cointet, J.P., Merckle, P., Fleury, E.: Detecting global bridges in networks. *Journal of Complex Networks* **4**, 319–329 (2016)
- [13] Kashtan, N., Itzkovitz, S., Milo, R., Alon, U.: Mfinder tool guide. Department of Molecular Cell Biology and Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot Israel, Tech Rep (2002)
- [14] Leskovec, J., Sosič, R.: Snap: A general-purpose network analysis and graph-mining library. *ACM Transactions on Intelligent Systems and Technology (TIST)* **8**(1), 1 (2016)
- [15] Lobier, M., Siebenhühner, F., Palva, S., Palva, J.M.: Phase transfer entropy: a novel phase-based measure for directed connectivity in networks coupled by oscillatory interactions. *NeuroImage* **85**, 853–872 (2014)
- [16] Milo, R., Kashtan, N., Itzkovitz, S., Newman, M.E., Alon, U.: Uniform generation of random graphs with arbitrary degree sequences. *arXiv preprint cond-mat/0312028* **106**, 1–4 (2003)
- [17] Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U.: Network motifs: simple building blocks of complex networks. *Science* **298**(5594), 824–827 (2002)
- [18] Rosenblum, M., Pikovsky, A., Kurths, J., Schäfer, C., Tass, P.A.: Phase synchronization: from theory to data analysis. *Handbook of Biological Physics* **4**, 279–321 (2001)
- [19] Schreiber, T.: Measuring information transfer. *Physical Review Letters* **85**(2), 461 (2000)
- [20] Sporns, O., Chialvo, D.R., Kaiser, M., Hilgetag, C.C.: Organization, development and function of complex brain networks. *Trends in Cognitive Sciences* **8**(9), 418–425 (2004)
- [21] Sporns, O., Kötter, R.: Motifs in brain networks. *PLoS Biol* **2**(11), e369 (2004)
- [22] Stam, C.J., Van Straaten, E.: The organization of physiological brain networks. *Clinical Neurophysiology* **123**(6), 1067–1087 (2012)
- [23] Tononi, G., Edelman, G.M., Sporns, O.: Complexity and coherency: integrating information in the brain. *Trends in Cognitive Sciences* **2**(12), 474–484 (1998)
- [24] Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Joliot, M.: Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *NeuroImage* **15**(1), 273–289 (2002)
- [25] Van Mieghem, P.: *Graph Spectra for Complex Networks*. Cambridge University Press (2011)
- [26] Zhigulin, V.P.: Dynamical motifs: building blocks of complex dynamics in sparsely connected random networks. *Physical Review Letters* **92**(23), 238,701 (2004)

Functional Reconstruction of Dyadic and Triadic Subgraphs in Spiking Neural Networks

Myles Akin, Alex Onderdonk, Rhonda Dzakpasu and Yixin Guo

Abstract Neural networks reconstructed from measurement data are known to exhibit various forms of nonrandom structures, including subgraph motifs and small-worldedness. It has been suggested such nonrandom structures are critical for neural information-processing; however, it is unclear how the topological structure of anatomical networks influences the reconstruction of functional networks. To better understand the importance of such nonrandom structures, we study how dyadic and triadic subgraphs are preserved during the reconstruction. We use a model-free information-theoretic measure, transfer entropy, to quantify the directional associations of pairwise neuronal activity. We employ multiplex networks to compare how dyadic and triadic subgraphs differ from structural to functional networks, with particular attention to recurrent connections. We find that certain structural subgraphs have more influence on the topology of the functional network than others.

1 Introduction

In neuroscience, abstract complex networks have been used to model the structure of neural tissue at both the macroscopic and microscopic scale [3, 10, 12]. Neuronal networks are typically classified as either structural or functional networks. The former type represents anatomical or synaptic connections between brain regions (macro) or individual neurons (micro). We focus on the microscopic level as much of the brain's information-processing and storage capacity is thought to arise from its synaptic connections and the structure they determine [12]. As such, it is necessary

Myles Akin (e-mail: mga28@drexel.edu) · Alex Onderdonk (e-mail: alo39@drexel.edu) ·
Yixin Guo e-mail: yixin@math.drexel.edu*✉
Drexel University, Department of Mathematics

Rhonda Dzakpasu (e-mail: rd259@georgetown.edu)
Georgetown University, Department of Physics, Department of Pharmacology and Physiology,
* corresponding author.

This work is supported by NSF under grant DMS-1226180 to Yixin Guo.

to identify important features of this structure and the roles they play in information-processing and storage. For this reason, it is of interest to study directed subgraphs and in particular motifs within a network.

Motifs are directed subgraphs of the network that occur more often than would be seen in a random network. Motifs are classified by the number of nodes they contain. We investigate dyadic (2-node) and triadic (3-node) motifs in particular for their biological relevance. Recurrent (reciprocal) connections have been hypothesized to allow for the storage of large amounts of information in neural circuits [2]. Such connections can be represented by the bidirectional dyadic directed subgraph within a structural network. At present, it is not clear what percentage of connections in structural neural networks are recurrent. Anatomical studies have suggested that they could be locally 100% connected thus having high recurrence whereas electrophysiological studies approximate reciprocal connections in only 10% of neuron pairs. Recurrent edges have also been found to be overrepresented in higher-order network motifs [9, 11, 12], particularly in triadic motifs which have been studied in biological neural networks [8, 10, 11]. Certain triadic motifs have also been found to have interesting functional roles and influence over local communication through clustering effects in neuronal networks [4, 7]. **Fig. 3** in section 5.2 shows all possible three-node subgraphs.

By contrast, functional networks are built from statistical dependencies of activity in brain regions or individual neurons as measured by subjecting fMRI, EEG, and microelectrode array (MEA) data to dependence measures including mutual information, coherence, and transfer entropy. Depending on the method, functional networks can be undirected or directed. Functional networks have been shown to exhibit many nonrandom features such as small-worldedness [3, 14], well-defined community structure [10], hubs [10, 13], and motifs [8, 11]. These functional networks derive their structure from the underlying anatomic or synaptic configuration; however, the nature of this influence is unclear.

At the microscopic scale, information is transferred from presynaptic to postsynaptic neurons via action potentials, also called spikes. Therefore functional networks, being built from spike time dependencies, represent information flow and processing by structural neural networks. Given this fact along with the hypothesized importance of recurrent connections in information-processing and storage, it is of interest to study how well these subgraph connections are captured in functional network studies. By modifying the percentage of recurrent connections in a simulated structural network, we can find how well they are captured compared to single-direction connections and whether a bias exists favoring recurrent connections in functional reconstruction. To study the change in topology from the structural to the functional network, we use a multilayer network [1]. Single layer networks, known as monoplex networks, are limited in that their edges represent single types of interaction. In contrast, multilayer networks have been increasingly used to allow for multiple aspects of node interaction where each layer uses a different type of edge. Multilayer networks have found extensive application in areas of study such as epidemiology, social, economic and biological interactions. Here, we apply multilayer network analysis to study the relationship between structural and functional networks and in particular to

determine how the former influences the development of the latter. Specifically, we focus on the reconstruction of dyadic and triadic subgraphs in general to provide a foundation for future work distinguishing motifs.

2 Network Model

We set up networks of 100 neurons, 20 inhibitory and 80 excitatory, based on the Izhikevich spiking model. We start with a regular, fully recurrent (i.e. undirected) network, then we randomly select edges and remove one direction to study how the proportion of non-recurrent edges affects functional reconstruction. We reduce the percentage of recurrent edges from 100% to 20%, creating a pseudo-regular network. Alternatively, we create a pseudo-small-world network by rewiring some edges in the original regular, fully recurrent network before implementing the aforementioned recurrent edge-reduction process.

2.1 Neuron Models

We use the Izhikevich neuron model as it allows for a wide variety of spiking options while maintaining computational simplicity. The Izhikevich model is a resetting spiking model of the neuron voltage. When the potential hits a peak, it resets to a level below threshold. We choose two basic neuronal parameter sets for our network: regular-spiking excitatory pyramidal cells and fast-spiking inhibitory interneurons [6]. The pyramidal cells are given by

$$100\dot{v} = 0.7(v + 60)(v + 40) - u + I_E$$

$$\dot{u} = 0.03(-2(v + 60)) - u$$

When the voltage is greater than or equal to 35mV, v is reset to -50mV and u is set to $u + 100$. I_E is the summation of the synaptic inputs into the excitatory cells and an external random Poisson excitatory input. For inhibitory interneurons, we use the fast-spiking model given by

$$20\dot{v} = (v + 55)(v + 40) - u + I_I$$

$$\dot{u} = 0.2(U(v) - u)$$

For this particular model, when $v \geq 25$ mV, v resets to -45mV. The adaptation variable u depends on a function $U(v)$. This function $U(v)$ is dependent on a threshold value of v_b , which we set to -55mV. If $v \geq v_b$ then $U(v) = 0.025(v - v_b)^3$, otherwise $U(v) = 0$. I_I is the summation of the synaptic inputs into the inhibitory cells and a random Poisson excitatory input which has a frequency of 10Hz. We use a simple exponential decay model for both excitatory and inhibitory synapses given by

$$S_X = v_X e^{\frac{t-t_X-t_k}{\tau}}$$

where $X \in \{E, I\}$. v_X is the maximum voltage increase delivered by the synapse to the postsynaptic cell. For all synapses, the decay constant τ was set to $3ms$. The delay time, t_X was set to $5ms$ for excitatory synapses and $1ms$ for inhibitory synapses. The t_k variable represents the time that a spike occurs in the presynaptic cells.

2.2 Network Construction

We arrange 100 neurons, 80 excitatory and 20 inhibitory, at random on a 10×10 grid initially with undirected connections between any pair of neurons separated by distance $\sqrt{2}$ or less. This gives a total of 342 undirected edges among interior neurons (eight incident edges), boundary neurons (five incident edges), and corner neurons (three incident edges). This regular lattice network structure does not have periodic boundaries and thus does not allow for propagation of looping activity through the network.

To test how recurrent edges affect the functional reconstruction of the underlying structural network, we randomly eliminate one direction of some undirected edges. To do this, we set a probability p_r which determines the proportion of undirected edges in the network which are selected to become directed. Another probability p_d determines which direction is deleted from each chosen undirected edge. When $p_r = 0$, the network remains completely undirected and when $p_r = 1$ the network has no recurrent edges. We study a variety of cases for p_r and set $p_d = 0.5$ so that the direction of deletion has no preference. It is interesting to note that the network is acyclic when $p_r = 1$ and $p_d = 0$ or 1 .

We also construct small-world networks using the algorithm proposed by Watts and Strogatz [14] since biological neural networks exhibit this structure [3, 14]. We step the rewiring probability, p_{rw} from 0 to 1 using steps of 0.2. For $p_{rw} = 0$, the network retains its lattice structure and for $p_{rw} = 1$ the network is completely random. In each case of small-world rewiring, we subsequently apply our recurrent edge reduction algorithm just as before.

3 Spike Train Analysis

We simulate a spiking network of Izhikevich neurons to obtain spike time-series. We then use the model free, high order transfer entropy method introduced in [5] to obtain dependencies between neuron pairs using the spike time-series. These measures are then thresholded at multiple values to produce functional network reconstructions. We use other model-free methods including coherence to discover dependencies between neurons for comparison, but focus on transfer entropy in the current paper due to page limitations.

3.1 Higher Order Transfer Entropy

Transfer entropy (TE), a standard tool for functional network reconstruction, identifies directed functional connections by comparing the spike-trains in 1ms time bins among a group of recorded neurons. For a given pair (i, j) of neurons, TE is a measure between zero and one which is greater in magnitude when including the spiking history of neuron j better allows for an accurate prediction of the spiking behavior of neuron i . We adopt a version of higher-order transfer entropy (HOTE) introduced in [5], which accommodates TE computation over a range of delays between two spike trains as well as over a range of orders; that is, the lengths of the spiking histories observed for neurons i and j . The HOTE formula is given by

$$TE_{j \rightarrow i} = \sum p(i_{t+1}, i_t^{(k)}, j_{t+1-d}^{(l)}) \log_2 \frac{p(i_{t+1} | i_t^{(k)}, j_{t+1-d}^{(l)})}{p(i_{t+1} | i_t^{(k)})} \tag{1}$$

where i_t, j_t give the states of neurons i and j (1 for the presence and 0 for the absence of a spike) at time bin t . k and l are the fixed orders of neurons i and j , respectively, and d represents the time delay between the observed states of the two neurons, ranging from 0 to 30ms. We set $k = 5$ and $l = 5$, and we compute HOTE using the MATLAB toolbox developed by Ito’s group [5].

3.2 Thresholding

To evaluate the significance of the reported TE result, each neuron in the network takes its turn serving as the “center” of the network, and we compute the mean μ and standard deviation σ of all TE values corresponding to connections which involve the chosen neuron. Since our networks are all directed, this requires separate statistical calculations for the center neuron’s incoming and outgoing TE values. For a choice of parameter κ , we compute the outward ($j \rightarrow$) and inward ($j \leftarrow$) thresholds, $\gamma_{j \rightarrow}^{(\kappa)}$ and $\gamma_{j \leftarrow}^{(\kappa)}$, respectively, by

$$\gamma_{j \rightarrow}^{(\kappa)} = \mu_{j \rightarrow} + \kappa \sigma_{j \rightarrow}, \quad \gamma_{j \leftarrow}^{(\kappa)} = \mu_{j \leftarrow} + \kappa \sigma_{j \leftarrow},$$

and an edge exists in the functional network from neuron j to neuron i if and only if

$$TE_{j \rightarrow i} \geq \max \left\{ \gamma_{j \rightarrow}^{(\kappa)}, \gamma_{i \leftarrow}^{(\kappa)} \right\}.$$

None of the functional networks produced are expected to be exact reconstructions of the structural network. In particular, a false-positive is reported when the inferred edge exists but the actual synapse does not, while an existing synapse not detected by TE is a false-negative.

4 Multiplex Networks

Multiplex networks are a special type of multilayer network in which each layer contains an identical set of vertices and the only interlayer edges connect corresponding vertices between layers [1]. The intralayer edges represent different types of connections, in our case structural versus functional, between the vertices. For each structural network, we build multiple multiplex networks whose functional layers are determined by various threshold values κ . We treat all nodes as identical ignoring whether they are inhibitory or excitatory. We will further explore the effects of inhibitory and excitatory neurons on network reconstruction in a later paper.

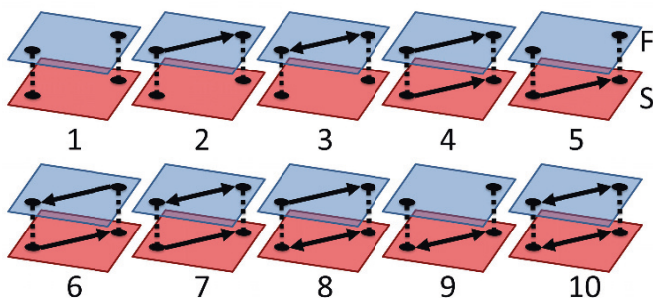


Fig. 1: The ten dyadic transformation subgraphs observed within multiplex networks containing functional (blue) and structural (red) layers. Note that numbers 1, 4, and 10 represent preservation of the structural connections.

We study how single-layer dyadic and triadic subgraphs transform from the structural to the functional layer, using multilayer subgraphs within these multiplex networks, see **Fig. 1** and **Fig. 6**. For dyadic subgraph transformations, we choose two vertices in the structural layer and the corresponding vertices in the functional layer. The resulting four-vertex subgraph defines a dyadic transformation. Triadic transformations are defined by the six-vertex multilayer subgraphs of the multiplex network. There are 10 dyadic transformations (**Fig. 1**) and 4096 triadic transformations. Several examples of the triadic transformation graphs are shown in **Fig. 6** in section 5.2. The rate at which these transformations are observed indicates the influence of the structural networks on their functional reconstruction.

5 Results

We study functional subgraph reconstruction of a simulated small-world structural neural network of Izhikevich neurons with $p_{rw} = 0.40$ and recurrent connection percentages of 100%, 80%, 60%, 40% and 20%. We produce ten-minute spike-trains as shorter data yields poor TE results [5]. The average firing rate decreases from 10Hz at 100% recurrent to 4Hz at 20% recurrent due in part to the reduced number of edges. We then use TE to determine pairwise dependencies which we

threshold at values of κ ranging from 0.1 to 2 with a step of 0.1. For each recurrent percentage, this results in 20 functional networks each of which we pair with the corresponding structural network. The resulting two-layer multiplex networks allow us to study transformations as defined in section 4 for dyadic and triadic subgraphs. We then determine which structural subgraphs have high influence over the functional reconstruction. As there are too many false-positives when $\kappa < 0.2$ and too many false-negatives when $\kappa > 0.8$, we focus our attention on examining the multiplex networks containing functional layers thresholded at $\kappa \in \{0.2, 0.5, 0.8\}$. We only present results for the networks with recurrent percentages of 60% and 20% due to space limitations. All simulations and network analysis are done using Cython and Python except that TE is done using Ito’s MATLAB Toolbox [5].

5.1 Dyadic Subgraphs

The transformation counts for dyadic subgraphs in these multiplex networks are presented in **Fig. 2**. Note we eliminate transformation 1 as it is trivial.

The sum of all the counts of transformations 4 through 7 is the total number of unidirectional dyadic subgraphs in the structural network. Subgraphs 6 and 7 involve the creation of a false-positive, functional edge in the opposite direction of the structural edge and both have negligible counts in our multiplex networks. This is expected as the influence of the structural edge dominates any residual dependence in the opposite direction. Transformations 2 and 3 give the majority of false-positives in the functional network. We will discuss further details about false-positives in the next section.

The sum of transformations 8 to 10 is the number of recurrent dyadic connections. The scarcity of transformation 9 implies that at least one direction of recurrent structural edges tend to be preserved in the functional reconstruction. Transformation 8 indicates that many recurrent connections lose one direction in the functional network, which implies that one neuron in the pair has greater influence over the transfer of information.

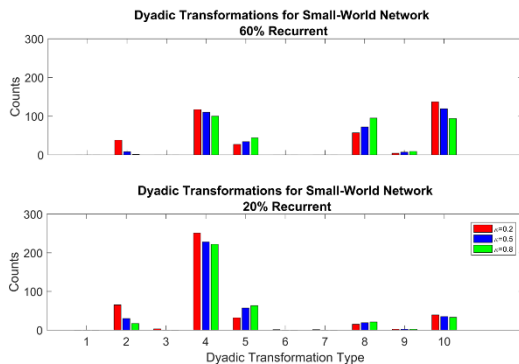


Fig. 2 Counts for each of the dyadic transformations from **Fig. 1** in functional networks with three threshold values. Note that we ignore the trivial transformation 1. All nodes are treated as identical.

The complete disappearance of a recurrent connection, transformation 9, is less frequent than that of a unidirectional connection, transformation 5, due to reciprocal information flow in the recurrent connections. We think that unidirectional connections are more prone to information loss, therefore recurrent connections may be more biologically suitable for maintaining information flow throughout the network.

5.2 Triadic Subgraphs

For triadic subgraphs (shown in Fig. 3), we first study the count comparison, shown in Fig. 4, between the structural and functional networks as it guides what warrants investigation in the subgraph transformation analysis. Triad 11 decreases dramatically as the percentage of recurrent connections is reduced from 60% to 20%, as demonstrated by the yellow bars shown in Fig. 4. We also notice, at both recurrent percentages, that the count of triad 11 in the functional network, regardless of threshold, is around half or less than that of the structural. This is due to the same tendency of recurrent connections to transform to unidirectional connections seen in the dyadic analysis. This indicates transformations of triad 11 are important in shaping the functional topology. Triads 7 and 8 have similar topological structure and exhibit comparable counts in the structural network; however, they present very different counts in the functional network. We observe that, in both 60% and 20% recurrent networks, the counts of triads 4 and 5 increase dramatically for low thresholds due to the reduction of higher indexed triads through false-negatives to lower indexed triads. This also occurs for higher thresholds in the 60% recurrent case.

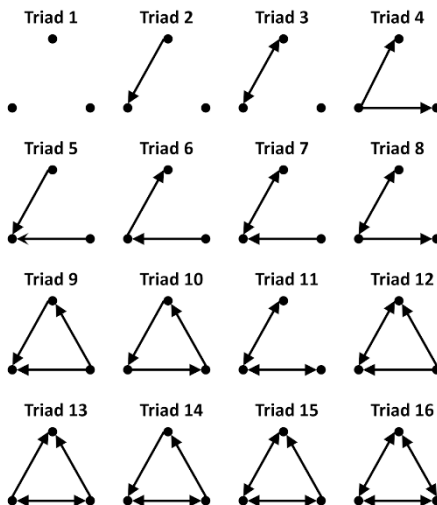


Fig. 3 This figure shows all possible 3-vertex triadic subgraphs of a directed network. Subgraphs that occur at statistically high rates are called motifs. It is common in literature to ignore subgraphs 1,2 and 3; however, we include them as possible transformation results.

Due to the extremely small number of occurrences, we could not glean much information about transformations of triads 9, 10, and 12 through 16. They are not of

interest of the current study; however, triad 9 shows a higher count in the functional network in general and plays an important role in “triangle completion” due to false-positive edges. We will discuss “triangle completion” in detail later. We ignore counts of triads 1, 2, and 3 as they are relevant only to transformations.

Fig. 4 Counts of the triadic subgraphs in small-world structural networks (shown by yellow bar) and the derived functional networks using three different threshold values. Subgraph number corresponds to those in Fig. 3. Triads 1 through 3 are ignored. All nodes are treated as identical.

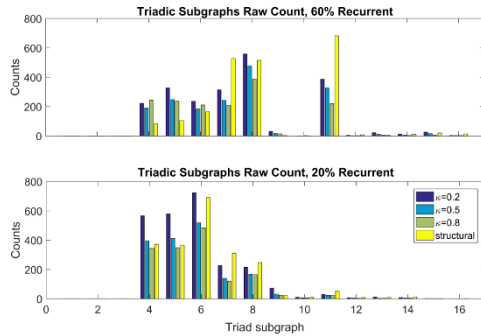
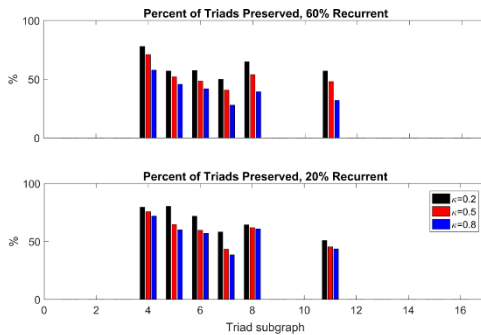


Fig. 5 Preservation rates of triads 4-8 and 11 in multiplex networks constructed from three threshold values. Due to limited number of triads 9, 10, and 12-16, these are ignored.



Subgraph preservation serves as an indicator of functional importance. The percentage of structural triads preserved in the functional network is given in Fig. 5. We notice that triad 4 is preserved at a higher rate than 5 and 6. These are the triads containing exactly two unidirectional and no recurrent edges. Triads 7 and 8 both contain a recurrent connection and exactly one unidirectional connection; however, triad 8 is preserved at a higher rate than 7. The high preservation of triads 8 and 4, both of which exhibit two outward edges emanating from the “middle” vertex, suggests that this structure allows for more robust local communication.

Triad 11 is prominent in the 60% recurrent structural network and transforms to triads 1-8, 9, 13 and 15, but with strong preference to triads 8 and 5. For $\kappa = 0.2$, triad 11 transforms 51% of the time to triad 8 and 27% to triad 5. All other transformations contributed less than 5% each, in particular, 2% by triad 7. Triad 11’s tendency to become triad 8 and the high preservation of triad 8 indicate the latter’s importance in shaping the functional network and local communication. As the threshold increases, transformations to triads 5 and 8 continue to dominate, though to a lesser extreme.

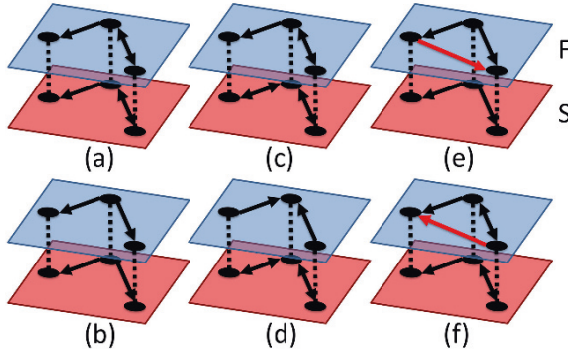


Fig. 6: Examples of triadic transformations. (a) and (b) show triad-preserving transformations for triads 8 and 4, respectively. (c) shows the transformation from triad 11 to triad 8 and (d) is the transformation from triad 11 to triad 5. (e) transforms triad 4 to 9 and (f) transforms triad 8 to 13. Both (e) and (f) demonstrate triangle completions (red edges are false-positives). All nodes are treated as identical.

These results manifest in the 20% recurrent network as well, though to a lesser extent in light of the reduction in recurrent connections. We also note the importance of transformations involving structural triads 7 and 8, but we leave this for the extended paper.

We now focus on the role of false-positives which can have profound influence on the topology of the functional network. We define “triangle completions” to be transformations that take a triad with exactly two pairs of connected nodes in the structural network to a triad in which all three nodes are connected in the functional network, regardless of direction. See **Fig. 6** (e) and (f) for examples. We find that nearly all false-positives arise through triangle completion. In the 60% recurrent network, we observe that triangle completions mainly involve the transformation of triad 8 to triads 9, 13 and 14 as well as that of triad 11 to triads 13, 14 and 15. In the 20% recurrent case, the majority of triangle completions concerns the transformation of triad 4 to triad 9 and that of triad 8 to triad 13. This again implies the important role of certain structural subgraphs in shaping the topology of the functional network. We explore this phenomenon in depth in the extended paper.

6 Discussion

In this paper, we study how dyadic and triadic subgraphs in spiking neural networks are reconstructed in functional networks using higher order transfer entropy. We showed that recurrent connections in general preserved at least one direction of connection in the inferred functional network. This indicates that one neuron in a pair shows greater influence over the spiking activity of the other.

We found that certain triadic subgraphs display tendencies towards preservation or transformation. These dispositions significantly influence the structure of the func-

tional network and consequently its information-processing capacity. The prevalence of structural subgraphs may impact the classification of functional triads as motifs. A discussion of motifs is left to the extended paper.

To evaluate any statistical significance of the transformation counts, we will have to establish a null model consisting of two phases. We will first demonstrate that the functional reconstruction depends nonrandomly upon the structural network. This will require a comparison of our observed transformation counts to a statistical average of those across many trial multiplex networks with randomized structural layers. By keeping the functional layer fixed across these trials, we will detect overrepresented and underrepresented transformations in our multiplex network which implicate nonrandom effects from our initial structural network.

Having established the statistical significance of our reconstruction, we turn to the second null model phase in which we evaluate the significance of the particular false-positive/negative features of the transformations. To this end, we average across a number of trials where the structural layer is fixed in its original form but the functional layers each contain randomized false-positive/negative edges. The proportion of false-positives/negatives is fixed across trials, so again detection of overrepresented or underrepresented transformations with these features implies significant influence from the underlying structure.

It has been shown in [7] that the position of inhibitory neurons within a triadic subgraph affects spiking behavior and consequently dependence within the triad. In this paper, we did not consider the position of inhibitory neurons in dyadic or triadic subgraphs. As such, further investigation into its effects on the functional reconstruction of these subgraphs will be required.

In our study, we used a fairly narrow Gaussian distribution of synaptic strengths which may not be realistic. To further our study, we can use either a wider Gaussian distribution or use spike-timing dependent plasticity (STDP) to learn synaptic strengths in order to obtain a less narrow distribution of synaptic strengths. This in turn will influence reconstruction of the structural network through an increase or decrease in dependence. It would also be of further interest to use STDP to study whether there is a learned preference for recurrent connections and certain triadic subgraphs.

References

- [1] Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C.I., Gómez-Gardeñes, J., Romance, M., Sendiña-Nadal, I., Wang, Z., Zanin, M.: The structure and dynamics of multilayer networks. *Physics Reports* **544**(1), 1–122 (2014)
- [2] Brunel, N.: Is cortical connectivity optimized for storing information [quest]. *Nature neuroscience* (2016)
- [3] Downes, J.H., Hammond, M.W., Xydias, D., Spencer, M.C., Becerra, V.M., Warwick, K., Whalley, B.J., Nasuto, S.J.: Emergence of a small-world functional network in cultured neurons. *PLoS Comput Biol* **8**(5), e1002522 (2012)
- [4] Guo, D., Li, C.: Stochastic and coherence resonance in feed-forward-loop neuronal network motifs. *Physical Review E* **79**(5), 051921 (2009)

- [5] Ito, S., Hansen, M.E., Heiland, R., Lumsdaine, A., Litke, A.M., Beggs, J.M.: Extending transfer entropy improves identification of effective connectivity in a spiking cortical network model. *PloS one* **6**(11), e27,431 (2011)
- [6] Izhikevich, E.M.: *Dynamical systems in neuroscience*. MIT press (2007)
- [7] Li, C.: Functions of neuronal network motifs. *Physical Review E* **78**(3), 037,101 (2008)
- [8] Perin, R., Berger, T.K., Markram, H.: A synaptic organizing principle for cortical neuronal groups. *Proceedings of the National Academy of Sciences* **108**(13), 5419–5424 (2011)
- [9] Rubinov, M., Sporns, O.: Complex network measures of brain connectivity: uses and interpretations. *Neuroimage* **52**(3), 1059–1069 (2010)
- [10] Shimono, M., Beggs, J.M.: Functional clusters, hubs, and communities in the cortical micro-connectome. *Cerebral Cortex* **25**(10), 3743–3757 (2015)
- [11] Song, S., Sjöström, P.J., Reigl, M., Nelson, S., Chklovskii, D.B.: Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS Biol* **3**(3), e68 (2005)
- [12] Sporns, O.: *Networks of the Brain*. MIT press (2010)
- [13] Timme, N.M., Ito, S., Myroshnychenko, M., Nigam, S., Shimono, M., Yeh, F.C., Hottowy, P., Litke, A.M., Beggs, J.M.: High-degree neurons feed cortical computations. *PLoS Comput Biol* **12**(5), e1004,858 (2016)
- [14] Watts, D.J., Strogatz, S.H.: Collective dynamics of small-world networks. *nature* **393**(6684), 440–442 (1998)

Modeling and Extending Ecological Networks Using Land Similarity

Gianni Fenu, Pier Luigi Pau and Danilo Dessì

Abstract Complex network analysis is being applied on topological models of ecological networks, to extrapolate their advanced properties and as part of the activity of land management. Commonly employed methods tend to focus on single target species. This is satisfactory for cognitive analysis, but the limited view provided by these models results in a lack of general information needed for land planning. Similarity scores computed for pairs of nature protection areas are proposed as a building block of a general model to address this shortcoming.

1 Introduction

Nature protection areas are established to protect endangered habitats and species from possible destruction due to the effects of increasing urbanization. Over the decades, policies have shifted toward the creation of ecological networks with a focus on the preservation of biodiversity. In the European Union, the establishment of a wide ecological network is the main goal of the Natura 2000 project.

Current methods to build graph models for ecological networks keep the focus on a species of interest. The resulting graphs are useful to perform quantitative analysis with respect to the target species, but the analysis of a large number of graphs is necessary to assess general properties of the network. In this paper, similarity scores between nature protection areas are proposed as a building block for graph models with a higher degree of generality, and different approaches are evaluated according to their aptness to the process of proposing network modifications.

The paper is organized as follows: in Section 2, basic information is provided concerning ecological networks, their graph representations, and goals of analysis. In Section 3, the aptness of available data on Natura 2000 sites to this study is

Gianni Fenu (e-mail: fenu@unica.it) · Pier Luigi Pau (e-mail: pierluigipau@unica.it) · Danilo Dessì (e-mail: daniilo_dessi@unica.it)
Department of Mathematics and Computer Science, University of Cagliari, Via Ospedale 72, 09124 Cagliari, Italy

discussed. In Section 4, the sites located in Sardinia are presented as a case study, and similarity-based graph models are introduced; three approaches to their construction are provided. In Section 5, correlations are sought between graph models, in order to determine which is most useful for land management and planning. Lastly, in Section 6, conclusions are drawn and possibilities for future work are discussed.

2 Ecological Networks and Graph Models

The expansion of human activities in every sector has caused radical modifications in land use, with a destruction of portions of habitats, and a fragmentation of those still in place. To protect habitats and species at risk of extinction, nature protection areas have been created. As the effectiveness of these areas is strongly reduced if habitat patches are too small or too distant from similar ones, policies have converged toward the creation of ecological networks, with each area contributing to large-scale preservation goals, and more endeavors to preserve the possibility of migration of species, in order to protect biodiversity [12]. Where necessary, migration can be encouraged by the establishment of man-made ‘habitat corridors’, either contiguous or in the form of ‘stepping stones’, i.e. sets of disconnected patches.

In the European Union, an ecological network is maintained as part of the project denominated “Natura 2000”. Its elements are sites designated as Special Protection Areas (SPA), as defined in the EU Birds Directive (2009/147/EC), and Special Areas of Conservation (SAC), as defined in the EU Habitats Directive (92/43/EEC); the latter are preliminarily designated as Sites of Community Interest (SCI). The boundaries of a SPA can overlap with those of SACs or SCIs, and vice versa; sites of the same category can be adjacent to one another.

The maintenance of ecological networks is becoming an essential aspect of land management and planning: local administrations are directly involved when Natura 2000 sites are in their jurisdiction, and can be affected by the presence of neighboring sites as well, due to their involvement in the possible creation of habitat corridors. Administrations are involved with the identification of threats and the proposal of a course of action to address standing issues with proper land management planning, which requires the consideration of several technical, regulatory, and political aspects. Tools to perform quantitative analysis on models representing an ecological network could make for an important contribution to the solution of these problems.

In analogy with many other kinds of networks and complex systems, a mathematical model for ecological networks is generally based on a graph, consisting of a set of nodes and a set of edges. A node may represent a site or habitat patch, depending on the desired scale, while edges represent connections. Graph models are built to represent functional connectivity with respect to a target species [11], while structural connectivity is analyzed with Geographic Information System tools. Quantitative analysis can uncover advanced properties of a network, which are not easily devised from its geographical map. Moreover, it enables comparison of graph models built for different target species in the same area, for a single target species in different areas, or representing different proposals for network modification.

Complex network analysis involves the study of statistical properties of graphs, related to node degree, shortest path length, and other features. Among the most commonly used indices are the clustering coefficient, related to the degree of redundancy of links; and the betweenness centrality index, often used to rank nodes by importance, according to their occurrence in shortest paths. The meaning of indices ought to be investigated according to each kind of real-world network being represented [3]; interpretations of several complex network indices have been proposed for ecological networks [4]. Global indices can be used as a measure of ‘health’ of the network, and local indices may assist in identifying vulnerabilities in topological networks [6], often associated to resiliency to node removal [5]. In general, the comparison of indices of a given network with those of modified versions is useful to predict the effect of modifications.

3 Similarity of Natura 2000 Sites

In order to collect data on the habitats and species found within the area, and to evaluate the impact of changes over time, reports are filed periodically for each Natura 2000 site. Information is gathered on-site and written to a data base conforming to a Standard Data Form, released with Commission Implementing Decision 2011/484/EU. Each Natura 2000 site is made up of patches of different habitat types, and each patch may host a different set of species. However, habitats and species found within a Natura 2000 site are reported to be present in the site, but no explicit relationship is established between each species and the habitat patch where it is found. This is sensible for the purposes of the Natura 2000 project, but it has a drawback in the fact that the knowledge of which habitat type is ideal for each species is not stored; rather, it is assumed to be part of expert knowledge or found in external documents. As a consequence, it is not straightforward to represent constraints that apply when proposing modifications.

To address these problems at least partially, it is possible to represent each site as a vector and compute similarity scores of these vectors, thus estimating a similarity score for pairs of sites. A minimum score between a pair of sites can be a prerequisite for the proposal to add an edge to the network. Adopting a similarity score taking values from 0 to 1 (where 1 is associated with pairs of identical vectors), such as the Jaccard coefficient or cosine distance, makes it easy to choose a threshold value.

The reported presence of species and that of habitats are two viable choices to build vectors representing Natura 2000 sites, using only data collected for the Natura 2000 project. A third viable approach is given by computing the intersection of sites with land use data from the CORINE program (Coordination of Information on the Environment). For this study, this was done using the open source QGIS [8] software suite. It is notable that CORINE land use data is available for areas outside of Natura 2000 sites; this is important to be able to combine graph-theoretic approaches and GIS functions [7], to determine whether it is possible to establish contiguous corridors. Land use types are categorized in a hierarchical manner with five levels of increasing detail. Only the first three levels of detail were used, as the fourth and

fifth were not available consistently; thus, a vector for each site was built by counting land patches intersecting the site, corresponding to each 3-digit code.

4 Case Study

In this work, the subset of Natura 2000 sites found in Sardinia is presented as a case study. At the time of writing, there is a total of 124 sites counting those designated as SPA and SCI; however, seven sites were excluded from this study because of unavailable land use data. If the boundaries of a SPA and a SCI overlap, two nodes are created, but they are considered to be at zero distance from each other. In all graph instances, an edge is not drawn between a pair of nodes if their approximate distance (calculated between borders on a map projection, using SQLite with the Spatialite extension) is greater than a set threshold (30 Km). The resulting network has 117 nodes, each corresponding to a Natura 2000 site. When all pairs of nodes where the geographical distance is up to 30 Km are linked, there is a total of 850 edges in the graph model. This shall be referred to as the *raw-distance graph* (Figure 1a).

A *single-species graph* is a model built to represent the state of the network with respect to a single species. In order to build a single-species graph, node pairs are linked with an edge if their distance is below the threshold and the presence of the species has been reported in both sites. Single-species graphs were built for all species listed in Annex II to Directive 92/43/EEC, plus others for which a species code consistent across site reports was given; an example is in Figure 1b. The open source Cytoscape suite (version 3.4.0) was used for graph visualization [9] and analysis, through the native NetworkAnalyzer plugin [2].

To represent the state of the network from a more general point of view, it is possible to build a graph instance based on site similarity, which shall be generally referred to as a *similarity-based graph*. This corresponds to a modification of the raw-distance graph, with the removal of edges that link node pairs with a similarity score below a set threshold. Clearly, different ways to compute similarity scores result in different graphs. In this study, three graphs are built for analysis, each based on Jaccard coefficients calculated on different vector representation of sites: the set of species reported to be in a site (*species-set graph*), the set of habitats found in the site according to Natura 2000 project data (*habitat graph*), and the set of level 3 land use codes according to the CORINE program (*land-use graph*). A similarity score of 0.5 shall be used as a threshold for all similarity-based graphs. In fact, recalling that the raw-distance network has 850 edges, similarity scores of 0.6 and above turn out to be strong requirements, removing over 85% of edges in all cases (Table 1).

5 Analysis of Edge Hit Rates and Complex Network Indices

The analysis of a single-species graph, with the extraction of its indices, is meant to give insight on the state of the network for the purpose of conservation of that species. As land management proposals may be reflected by modifications on the graph model, the improvement of indices according to set goals can act as a criterion

Table 1: Number of edges in similarity-based graphs of Natura 2000 sites in Sardinia

Minimum similarity	Land use-based	Habitat-based	Species-based
0.0 (raw-distance)	850	850	850
0.4	360	295	198
0.5	232	205	117
0.6	104	120	53

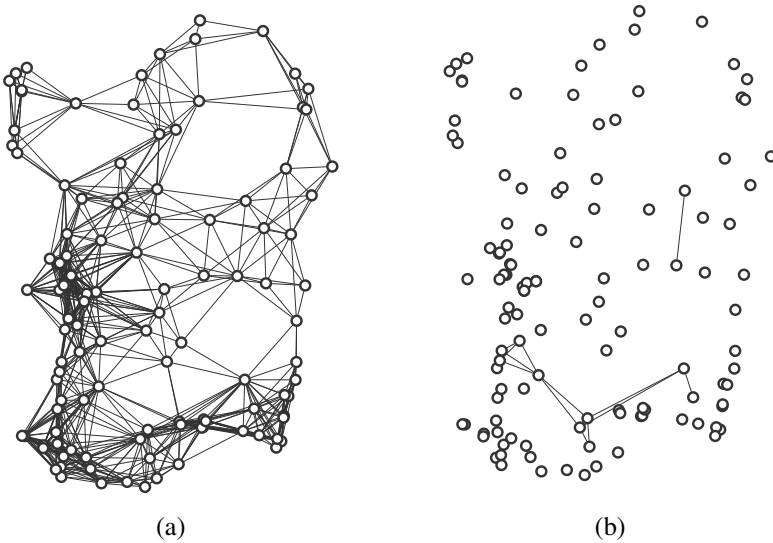


Fig. 1: Graph models of Natura 2000 sites in Sardinia. (a) Raw-distance graph. Edges link pairs of nodes with a geographical distance up to 30 Km between boundaries. The position of each node roughly corresponds to the coordinates of the site centroid. (b) Single-species graph for *Cervus elaphus corsicanus* (species code 1367). For comparison purposes, all nodes from the raw-distance graph are represented; technically, only linked nodes are part of this graph.

to identify favorable modifications. In many complex network applications, the set of nodes is to be kept unmodified; assuming the network is initially connected, proposed modifications can fall into one of three categories [1]:

- Addition of edges (also referred to as ‘updating’; proposed edges are referred to as ‘virtual edges’). Assuming that adding links in the real-world network has a cost, this problem is related to that of finding a set of new links which brings as great a benefit as possible, while respecting budget constraints.
- Removal of edges (‘downdating’). Assuming that the network has some degree of redundancy, this problem is related to that of finding a set of edges that can

be removed to decrease maintenance costs, while affecting the efficiency of the network as little as possible. The network as a whole should not be disconnected.

- Rewiring, i.e. removing and subsequently adding one or more edges. This problem is related to that of improving the efficiency of a network, while avoiding an increase in maintenance costs.

In land management for ecological networks, an interesting problem may be to find a site to relocate part of the population of a species, among those where it has not been reported, while preserving or enhancing the emerging network effect; this can be done to improve network indices or to merge components which are not initially connected. In the graph model, this is reflected as an addition of nodes; this poses a few problems, most notably the identification of suitable candidate sites for node addition.

As previously mentioned, the Natura 2000 dataset was not designed with this problem in mind, hence it is not straightforward to suggest good candidate nodes to extend any given single-species graph. Newly connected sites should be within a set geographical distance from an already connected node, and should host the preferred habitat for the target species, or a suitable set of habitats for a temporary settlement of the species, if the node is to act as a ‘bridge’.

One of the methods to calculate site similarity scores may qualify as a way to formalize this criterion when data on habitat suitability is missing or incomplete. Then, similarity-based graphs become a useful tool to express this notion. In formal terms, a good candidate has the property that, in a similarity-based graph, it is adjacent to a node that is part of a connected component in the single-species graph. In symbols, let V be the full set of nodes representing Natura 2000 sites in the region of interest, and let $G_s = (V, E_s)$ be a similarity-based graph built on node set V with a suitable geographical distance threshold. Let $G' = (V', E')$ be a connected component in the single-species graph built on V for the target species ($V' \subseteq V$), with the same geographical distance threshold used for G_s . Then, if

$$i \in V', \quad j \in V, \quad j \notin V', \quad (i, j) \in E_s, \quad (1)$$

then $j \in V$ is a good candidate node, and (i, j) is a candidate edge to link j to G' .

Since there are more ways to build G_s , an interesting question is which similarity-based graph is best for the purpose of determining good node candidates. Intuitively, if edges in single-species graphs often appear as edges in G_s , then G_s should provide better candidates for graph updating. To measure the aptness of the similarity-based graphs built on Jaccard coefficients of species sets, habitats and land use codes, the three graphs were compared with 351 single-species graphs, using the same 30 Km distance threshold. Results for a few sample species are reported in Table 2, together with average rates. The three similarity-based graphs rank about the same way at a 34% average hit rate, with a slight disadvantage for the species-set graph at 31%.

These low hit rates do not show a clear winner among the three criteria under consideration for building similarity-based graph; not only that, but they suggest that there may be remarkable differences among the three graphs, which is confirmed by comparing them visually (Figure 2).

Table 2: Excerpt of the table of hit rates. For each species, the number of edges in the single-species graph is reported. Then, for each similarity-based graph, it is shown how many of those edges are present in the similarity-based graph (hits), and the corresponding rate. The last row reports the average of all hit rates for each similarity-based graph.

Species code	Edges	Land use-based		Habitat-based		Species-based	
		Hits	Rate	Hits	Rate	Hits	Rate
...							
6137	186	93	0.5	55	0.29570	24	0.12903
1367	15	9	0.6	8	0.53333	6	0.4
1373	8	6	0.75	2	0.25	2	0.25
...							
Average hit rate			0.33650		0.34341		0.31308

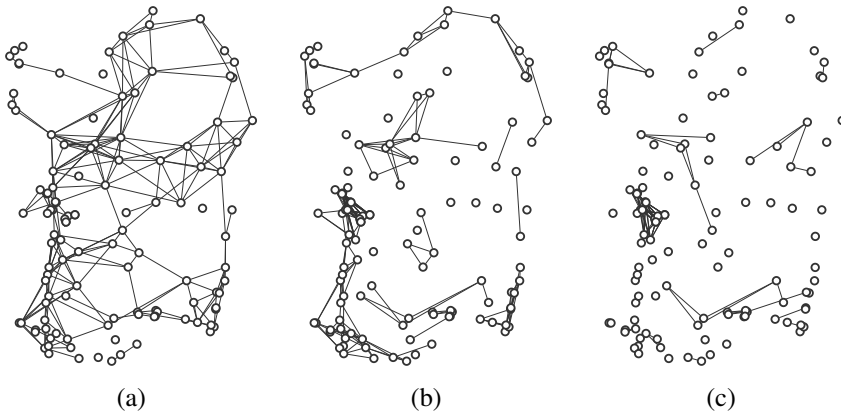


Fig. 2: Similarity-based graph models of Sardinian Natura 2000 sites, with a 0.5 similarity score threshold and a 30 Km distance threshold. (a) Based on CORINE land use codes. (b) Based on Natura 2000 habitat codes. (c) Based on species sets.

To establish whether these differences are significant, it is interesting to assess whether any pair of similarity-based graphs behave similarly with respect to hit rates; essentially, if the hit rate for a similarity graph G_s were high for the same species as that of another similarity graph G_t , it could be argued that G_s and G_t express a similar concept. To do so, Spearman correlation indices are calculated between pairs of columns reporting hit rates in Table 2. It is notable that, while no correlation is detected between the land use graph and the others, the species-set and habitat graphs appear to have a strong correlation, above 0.8 (Table 3). This is not only consistent with the fact that land use data originates from a different project; it confirms that

nearby sites with similar habitat sets also host similar sets of species, thus reinforcing the notion that the classification of habitats within the Natura 2000 project is more suitable to describe sites than land use codes are.

Table 3: Spearman correlation between sets of hit rates.

	Habitat-based	Species-based
Land use-based	0.08446	0.03489
Habitat-based		0.80397

To corroborate this notion, it is possible to extend the comparison to complex network indices calculated for nodes on the three similarity-based graph instances. Indices are calculated for nodes representing Natura 2000 sites on the three graph instances (see an excerpt in Table 4). The question is, for each index, whether a higher value calculated on a graph corresponds to a higher value calculated on another. Considered indices are node degree, closeness and betweenness centrality indices, clustering coefficient, and topological coefficient [10].

Table 4: Excerpt of the table of normalized betweenness centrality indices calculated for each node (site) on each similarity-based graph.

Site	Betweenness centrality index		
	Land-use	Habitats	Species-set
...			
ITB030034	0.01671	0.11557	0.04915
ITB030035	0.00014	0.04341	0.09402
ITB030036	0.01046	0	0.00641
...			

Then, correlation are sought between sets of values for the same index on the three possible pairs of graph instances, once again by calculating their Spearman correlation coefficients (see Table 5 and a visual representation in Figure 3). Contrary to hit rates, there is no value suggesting a strong correlation; however, a moderate degree of correlation can be identified between the species-set and the habitats graph for three measures (degree, topological coefficient and clustering coefficient), thus reinforcing the previous observations that these two graphs are more similar to one another, than the land-use graph is to either.

Table 5: Spearman correlation of various complex network indices, between pairs of similarity-based graphs.

Index	Land-use/Species	Land-use/Habitats	Species/Habitats
Betweenness centrality	+0.09309	+0.17446	-0.01905
Closeness centrality	+0.01001	-0.02426	+0.12268
Degree	+0.09172	+0.09961	+0.41257
Topological coefficient	+0.11214	+0.04271	+0.25071
Clustering coefficient	-0.02396	-0.10644	+0.28248

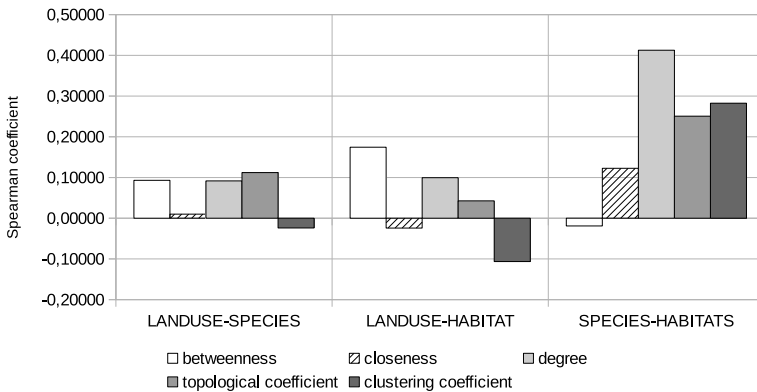


Fig. 3: Histogram representation of Spearman correlation of various complex network indices, between pairs of similarity-based graphs.

6 Conclusions and Future Work

Current methods to apply complex network analysis on ecological networks tend to focus on single species of interest, making it hard to evaluate and represent high-level properties of the network. The analysis of single-species graphs proves to be useful to assess the state of the network, but in the context of the Natura 2000 project in the European Union, methods for data collection and storage were not designed to assist researchers in proposing network modifications for its improvement. In this paper, the construction of graph models based on site similarity is proposed as a way to address this shortcoming. Multiple ways to build similarity-based graphs are discussed and compared; results suggest that land use data expresses different concepts than the species sets and habitat sets associated to each site. This represents a challenge for land managers seeking to detect or establish habitat corridors, since only land use data is available for land outside of Natura 2000 sites.

Future work will focus on an extension and application of the network updating problem on single-species models. The linking of nodes that are not initially con-

nected will be considered, subject to constraints based on site similarity and on the degree of contiguity of land use outside of Natura 2000 sites.

Acknowledgements This essay is written within the Research Program “Natura 2000: Assessment of management plans and definition of ecological corridors as a complex network”, funded by the Autonomous Region of Sardinia (Legge Regionale 7/2007) for the period 2015-2018, under the provisions of the Call for the presentation of “Projects related to fundamental or basic research” in year 2013, implemented at the Department of Mathematics and Computer Science of the University of Cagliari, Italy. Pier Luigi Pau gratefully acknowledges Sardinia Regional Government for the financial support of his PhD scholarship (P.O.R. Sardegna F.S.E. Operational Programme of the Autonomous Region of Sardinia, European Social Fund 2007-2013 - Axis IV Human Resources, Objective 1.3, Line of Activity 1.3.1.).

References

- [1] Arrigo, F., Benzi, M.: Updating and Datedating Techniques for Optimizing Network Communicability. *SIAM Journal on Scientific Computing* **38**(1), B25–B49 (2016)
- [2] Assenov, Y., Ramírez, F., Schelhorn, S.E., Lengauer, T., Albrecht, M.: Computing topological parameters of biological networks. *Bioinformatics (Oxford, England)* **24**(2), 282–284 (2008). DOI 10.1093/bioinformatics/btm554
- [3] Borgatti, S.: Centrality and network flow. *Social Networks* **27**(1) (2005)
- [4] Estrada, E., Bodin, Ö.: Using Network Centrality Measures to Manage Landscape Connectivity. *Ecological Applications* **18**(7), 1810–1825 (2008)
- [5] Iyer, S., Killingback, T., Sundaram, B., Wang, Z.: Attack Robustness and Centrality of Complex Networks. *PLOS ONE* **8**(4), e59,613 (2013)
- [6] Mishkovski, I., Biey, M., Kocarev, L.: Vulnerability of complex networks. *Communications in Nonlinear Science and Numerical Simulation* **16**(1), 341–349 (2011)
- [7] Pinto, N., Keitt, T.H.: Beyond the least-cost path: evaluating corridor redundancy using a graph-theoretic approach. *Landscape Ecology* **24**(2), 253–266 (2008)
- [8] QGIS Development Team: QGIS Geographic Information System. Open Source Geospatial Foundation (2009). URL <http://qgis.osgeo.org>
- [9] Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T.: Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research* **13**(11), 2498–2504 (2003). DOI 10.1101/gr.1239303
- [10] Stelzl, U., Worm, U., Lalowski, M., Haenig, C., Brembeck, F.H., Goehler, H., Stroedicke, M., Zenkner, M., Schoenherr, A., Koeppen, S., Timm, J., Mintzlaff, S., Abraham, C., Bock, N., Kietzmann, S., Goedde, A., Toksöz, E., Droege, A., Krobitsch, S., Korn, B., Birchmeier, W., Lehrach, H., Wanker, E.E.: A human protein-protein interaction network: a resource for annotating the proteome. *Cell* **122**(6), 957–968 (2005). DOI 10.1016/j.cell.2005.08.029
- [11] Urban, D.L., Minor, E.S., Treml, E.A., Schick, R.S.: Graph models of habitat mosaics. *Ecology Letters* **12**(3), 260–273 (2009)
- [12] Vimal, R., Mathevet, R., Thompson, J.D.: The changing landscape of ecological networks. *Journal for Nature Conservation* **20**(1), 49–55 (2012)

Part XI
Network Analysis

A graph-based, semi-supervised, credit card fraud detection system

Bertrand Lebichot, Fabian Braun, Olivier Caelen and Marco Saerens

Abstract Global card fraud losses amounted to 16.31 Billion US dollars in 2014 [18]. To recover this huge amount, automated Fraud Detection Systems (FDS) are used to deny a transaction before it is granted. In this paper, we start from a graph-based FDS named APATE [28]: this algorithm uses a collective inference algorithm to spread fraudulent influence through a network by using a limited set of confirmed fraudulent transactions. We propose several improvements from the network data analysis literature [16] and semi-supervised learning [9] to this approach. Furthermore, we re-designed APATE to fit to e-commerce field reality. Those improvements have a high impact on performance, multiplying Precision@100 by three, both on fraudulent card and transaction prediction. This new method is assessed on a three-months real-life e-commerce credit card transactions data set obtained from a large credit card issuer.

1 Introduction

Nowadays, e-commerce becomes more and more important for global trade: sales of goods and services represented more or less 2,000 billion dollars in 2014 and it was estimated that on 7,223 millions peoples on earth, 20 % were e-shoppers [14]. Part of the reasons of this success is easy online credit card transactions and cross-border purchases. Furthermore, most organizations, companies and government agencies have adopted e-commerce to increase their productivity or efficiency in trading products or services [4].

Bertrand Lebichot (e-mail: bertrand.lebichot@uclouvain.be) · Marco Saerens
(e-mail: marco.saerens@uclouvain.be)
Universite Catholique de Louvain, Place des Doyens 1, 1348 Louvain-la-Neuve, Belgium

Fabian Braun (e-mail: fabian.braun@worldline.com)
Worldline GmbH, R&D, Pascalstrasse 19, 52076 Aachen, Germany

Olivier Caelen (e-mail: olivier.caelen@worldline.com)
Worldline SA/NV, R&D, Chaussee de Haecht 1442, 1130 Bruxelles, Belgium

Of course, e-commerce is used by both legitimate users and fraudsters. The Association of Certified Fraud Examiners (ACFE) defines fraud as: "the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets" [8].

Global card fraud losses amounted to 16.31 Billion US dollar in 2014 and is forecast to continue to increase [18]. This huge number of losses has increased the importance of fraud fighting: in a competitive environment, fraud have a serious business impact if not managed, and prevention (and repression) procedures must be undertaken.

For those reasons e-commerce and credit card issuers need automated systems that identify incoming fraudulent transactions or transactions that do not correspond to a normal behavior. Data mining and machine learning offer various techniques to find patterns in data; here, the goal is to discriminate between genuine and fraudulent transactions. Such Fraud Detection Systems (FDS) exist and are similar to detection approaches in Intrusion Detection System (IDS). FDS use misuse and anomaly based approaches to detect fraud [15].

However, there are issues and challenges that hinder the development of an ideal FDS for e-commerce system [11]; such as,

- Concept drift: fraudsters conceive new fraudulent ways/methods over time. Furthermore, normal behavior also varies with time (peak consumption at Christmas for instance).
- Six-seconds rule [28]: acceptance check must be processed quickly as the algorithm must decide within six seconds if a transaction can be pursued.
- Large amount of data: millions of transactions occur per day whereas have to be analyzed and acceptance must be granted in seconds.
- Unbalanced data: frauds represents hopefully only less than 1% of transactions but predicting a pattern is harder with unbalanced dataset.

The presence of those challenges leads to high false alert rate, low detection accuracy or slow detection (see [1] for more details).

This work focuses on automatically detecting e-commerce fraudulent transactions using network (or graph) related features. Our work is based on a recent paper [28] which introduced an automated and field-oriented approach to detect fraudulent patterns in credit card transactions by applying supervised data mining techniques. More precisely, this algorithm uses a collective inference algorithm to spread fraudulent influence through a network by using a limited set of confirmed fraudulent transactions and take a decision based on risk scores of suspiciousness of transactions, card holder and merchants.

In this paper, several improvements from graph literature and semi-supervised learning are proposed. The resulting fraud detection method is tested on a three-months real-life e-commerce credit card transaction data set obtained from a large credit card issuer in Belgium.

The following questions are addressed:

1. Can we enhance graph-based existing FDS in terms of performance?
2. How can we make FDS as suitable for real application as possible?

3. Is semi-supervised learning [9] or feedback [11] useful for this Graph-based FDS?

Our approach takes into account various field/ground realities such as the six-second rule, concept drift, dealing with large datasets and unbalanced data. It also has been conceived in accordance with field experts to guarantee its applicability.

The rest of this paper is divided as follows: Section 2 introduces background and notation. Section 3 reviews related work. Section 4 details the proposed contributions. Experimental comparisons are presented and analyzed in Section 5. Finally, Section 6 concludes this paper.

2 Background and Notation

This section will first introduce some basic facts about fraud detection, since behavior of fraudsters has to be taken into account in the development of algorithms designed to counter them. Then some useful graph notation is reviewed.

2.1 Frauds

There are many fraud detection domains but internet e-commerce presents a challenging data mining task (see Section 1) because it blurs the boundaries between fraud detection systems and network intrusion detection systems.

As in many domains, profit-motivated fraudsters interact with the affected business. [2, 24] describes comprehensively this interaction: the fraudster can be internal or external to the business, can either commit fraud as a customer (consumer) or as a supplier (provider), and has different basic profiles. From this description, it comes out that professional fraudsters (as opposed to occasional ones) modus operandi changes over time. Therefore, fraud detection system algorithms should also adapt themselves to new behaviors. This is referred as "Concept drift": the constant change in fraudsters behavior.

2.2 Graphs

Consider a weighted directed graph or network, G , assumed strongly connected with a set of n nodes V (or vertices) and a set of edges E (or arcs, links) [6, 22]. The **adjacency matrix** of the graph, containing non-negative affinities between nodes, is denoted as \mathbf{A} , with elements $[\mathbf{A}]_{ij}$ (also written a_{ij}) ≥ 0 . The **natural random walk** on G is defined in a standard way. In node i , the random walker chooses the next edge to follow according to reference transition probabilities

$$p_{ij} = \frac{a_{ij}}{\sum_{j'=1}^n a_{ij'}} \quad (1)$$

representing the probability of jumping from node i to node $j \in Succ(i)$, the set of successor nodes of i . The corresponding transition probability matrix will be denoted as \mathbf{P} . In other words, the random walker chooses to follow an edge with a probability proportional to the affinity (apart from the sum-to-one normalization), therefore favoring edges associated to a large affinity. The matrix \mathbf{P} , containing the p_{ij} , is stochastic and is called the **reference transition matrix**.

3 Related Work

Credit-card Fraud detection received a lot of attention, but the number of publications available is limited. Indeed, credit card issuers protect data sources and most algorithms are produced in-house concealing the model's details [28].

As for any machine learning modeling process, two main approaches can be used: a supervised and an unsupervised scheme. Supervised learning uses labels (the observed prediction of an instance, here the fraud tag) to build the classification model, where unsupervised simply extracts clusters of similar data that are then processed. Common unsupervised techniques are peer group analysis [29] and self-organizing maps [30] while common supervised techniques are artificial logistic regression, neural networks (ANN) and random forests, meta-learning, case-based reasoning, Bayesian belief networks, decision trees, logistic regression, hidden Markov models, association rules, support vector machines, Bayes minimum risk and genetic algorithms. The reader is advised to consult [12] for more detail about credit card fraud detection, and [24] for a wider review on fraud detection.

According to [28], APATE was the only one to include network knowledge in the prediction models for fraud detection: This model first builds a tripartite graph (see below) and then extracts relevant risk scores for each node. [28] shows that this information, added to more conventional ones, increases the performances of the fraud detection system.

In this work, we follow the methodology of APATE [28] (which is described in this section, to make this paper self-contained), and propose several improvements in the next section. Other types of graph were also investigated (bipartite,...) but they did not provide better results and are therefore not presented here.

In particular, APATE starts with a set of time stamped, labeled, transactions. The goal is, of course, to fit a model to infer future fraudulent/genuine transactions. Furthermore, for each transaction of this dataset, the card holder (or user) and merchant (or retailer) is known. APATE thus create a tripartite adjacency matrix \mathbf{A}^{tri} (there are three type of node: transactions, card users and merchants) as follows:

$$\mathbf{A}^{\text{tri}} = \begin{pmatrix} \mathbf{0}_{t \times t} & \mathbf{A}_{t \times c} & \mathbf{A}_{t \times m} \\ \mathbf{A}_{c \times t} & \mathbf{0}_{c \times c} & \mathbf{0}_{c \times m} \\ \mathbf{A}_{m \times t} & \mathbf{0}_{m \times c} & \mathbf{0}_{m \times m} \end{pmatrix} \quad (2)$$

where $\mathbf{A}_{t \times c} = (\mathbf{A}_{c \times t})^T$ is an adjacency matrix where transactions are linked with their corresponding card holders, $\mathbf{A}_{t \times m} = (\mathbf{A}_{m \times t})^T$ is an adjacency matrix where

transaction are linked with corresponding merchants and $\mathbf{0} \dots \times \dots$ is a correctly sized matrix full of zeros. From \mathbf{A}^{tri} , transition matrix \mathbf{P} is derived (see Section 2.2).

A column vector $\mathbf{r}_0 = [\mathbf{r}_0^{\text{Trx}}, \mathbf{r}_0^{\text{CH}}, \mathbf{r}_0^{\text{Mer}}]^T$ of length equal to the total number of transactions (hence the superscript *Trx*), card holders (*CH*) and merchants (*Mer*) is also created. The vector is full of zeros, except for known fraudulent transactions where it is equal to one (and therefore is always zero for merchants and card holders). Finally, element *k* of a vector \mathbf{r}_0 is noted $[\mathbf{r}_0]_k$.

Then, in a convergence procedure similar to the *PageRank* algorithm [23], \mathbf{r}_0 is updated to spread the fraud label through the tripartite graph. This is known as a random walk with restart procedure (RWWR) [19]:

$$\mathbf{r}_k = \alpha \cdot \mathbf{P}^T \mathbf{r}_{k-1} + (1 - \alpha) \cdot \mathbf{r}_0 \tag{3}$$

where α is the probability to continue the walk and $(1 - \alpha)$ is the probability to restart the walk from a fraudulent transaction. This parameter could be tuned, but was fixed to 0.85 in the experimental comparisons (see [23]). The procedure diffuses the information about the transactions through the network.

Eq. 3 is iterated until convergence. Then, from \mathbf{r}_{kc} (where *kc* stands for *k* at convergence) $\mathbf{r}_{kc}^{\text{Trx}}$, $\mathbf{r}_{kc}^{\text{CH}}$ and $\mathbf{r}_{kc}^{\text{Mer}}$ can be extracted and considered as a risk measure for each transaction, card holder and merchant respectively.

As fraud detection models should adapt dynamically to a changing environment, this procedure is repeated several times, introducing a time decay factor. Each non-zero entry of \mathbf{A}^{tri} and \mathbf{r}_0 is modified to characterize transactions based on current and normal customers past behavior (see [28] for more details):

$$\begin{cases} [\mathbf{A}^{\text{tri}}]_{ij} \leftarrow e^{-\gamma \cdot t([\mathbf{A}^{\text{tri}}]_{ij})} & \text{or 0 if no relation} \\ [\mathbf{r}_0]_k \leftarrow e^{-\gamma \cdot t([\mathbf{r}_0]_k)} & \text{or 0 if no fraud} \end{cases} \tag{4}$$

where $t(\cdot)$ is the (scalar) time where transaction between *i* and *j* in matrix \mathbf{A}^{tri} occurred (or *k* for vector \mathbf{r}_0), and γ is a scalar set in such a way that the half-life of the exponential is: one day, one week and one month (i.e. elements are equal to 0.5 at half-life). For instance, if a transaction occurred two weeks ago, the corresponding element of \mathbf{A}^{tri} with week decay is equal to 0.25 and is $1/(2^{14})$ with day decay.

Therefore, for each transaction of our starting dataset, we have 12 new features: Transaction risk for transaction, card holder and merchant, each for four (no decay, day decay, week decay and month decay) time windows.

However, this procedure cannot be computed in less than a few minutes, which is not suitable with the six-seconds rule. Convergence on a graph with millions of nodes is expensive and is therefore daily re-estimated over night. Transactions made during the testing day are evaluated using the model trained on previous night. For card holders and merchants, the graph-based feature values are extracted (looked up) from the trained model, since they are likely to be part of the previous data.

Naturally, for the new transaction not part of the model, transaction-based features have to be estimated, which is done through the formula:

$$\text{score}(Trx_{i,k}) = \frac{1}{\sum_{j=1}^n p_{ji} + 1} \text{score}(Mer_i) + \frac{1}{\sum_{j=1}^m p_{jk} + 1} \text{score}(CH_k) \quad (5)$$

where $\text{score}(Trx_{i,k})$ stands for the new transaction score between merchant i and card holder k , $\text{score}(Mer_i)$ stands for the score of merchant i and $\text{score}(CH_k)$ stands for the score of card holder k . It represents the score of a new transaction l after one new iteration of Eq. 3 when this transaction is added to \mathbf{P} (with $p_{li} = 1$ and $p_{lk} = 1$). If a new transaction involves a new merchant and/or card holder, $\text{score}(Mer_i)$ and/or $\text{score}(CH_k)$ are set to zero accordingly.

Finally, those 12 new features (plus transaction-related features, see Table 1) are fed to a random forest classification model, as this model proved to perform well for the problem at hand, predicting fraudulent transaction [3, 12].

Table 1: Features used by the random forest classifier. First group are demographical features and second group are graph-based features. Notice that each transaction is linked with a card holder and with a merchant at a certain date: those information are only used to build the tripartite graph.

Variable name	Description
inBEL/EURO/OTH	Issuing region: Belgium/Europa/World
TX AMOUNT	Amount of transaction
TX 3D SECURE	Transaction used 3D secure
AGE	Age of card holder
langNED/FRE/OTH	Card holder language: Dutch/French/Other
isMAL/FEM	Card holder is Male/Female
isFoM	Card holder gender unknown
BROKER	Code of card provider
cardMCD/VIS/OTH	Card is a Mastercard/Visa/Other
01 Mer score	Merchant risk score (boolean, no time damping)
ST/MT/LT Mer score	Day/week/month decay merchant risk score (3 features)
01 CH score	Card Holder risk score (boolean, no time damping)
ST/MT/LT CH score	Day/week/month decay Card Holder risk score (3 features)
01 Trx score	Transaction risk score (boolean, no time damping)
ST/MT/LT Trx score	Day/week/month decay Transaction risk score (3 features)
TX FRAUD	Target variable: Fraud/Genuine

4 The Proposed Model

While showing good performance, APATE can be improved in various ways.

4.1 Dealing with hubs

From the literature, it is known that presence of hubs in a network can harm the classifier [17, 25, 26]: hubs are nodes having a high degree and are therefore neighbors of a large number of nodes. In our dataset, it corresponds to popular nodes such as popular online shops like Amazon (as an example, the dataset is anonymised). Those nodes tend to accumulate a high value of risk score since they are connected to a lot of transactions. A simple way to counterbalance this accumulation is to divide the risk score by the node degree after convergence. In general, it is possible to divide by any power of the node degree and/or by different powers for the three types of nodes of the tripartite graph (transactions, card holders and merchants). In practice however, we did not find any combination that significantly beats the simple divide-by-node-degree option (results are not reported here).

Furthermore, it allows us to make a link with the regularized commute time kernel which is $\mathbf{K} = (\mathbf{D} - \alpha\mathbf{A})^{-1}$ (where \mathbf{D} is the degree matrix) : element i, j of this kernel can be interpreted as the discounted cumulated probability of visiting node j when starting from node i (see [16, 21, 31] for details). The (scalar) parameter $\alpha \in]0, 1]$ corresponds to an evaporating or killed random walk where the random walker has a $(1 - \alpha)$ probability of disappearing at each step (therefore it has the same interpretation as for the RWR used in APATE, see Section 3). This method provided the best results in a recent comparative study on semi-supervised classification [16] and the second best results in another one [20]. In practice, the efficient implementation proposed in [21], Equation (22), for semi-supervised classification with the Regularized Commute Time Kernel is used and referred as RCTK.

4.2 Introducing a time gap

On the other hand, unlike in [28], the model cannot be based on past few days. Indeed, fraudulent transaction tags (the variable we want to predict) cannot be known with certainty without human investigators feedback. Moreover, since the fraudsters' modus operandi is known to change over time (see 2.1), it is not acceptable to built our model on old, less reliable (but fully inspected) data. However, it takes several days to inspect all transactions, mainly because it is sometime card holders that report undetected frauds. Of course, this makes our fraud detection problem harder [10].

In arrangement with field experts, we designed a real-life scenario containing three sets of data:

1. Training set: data where the transaction fraud labels can be taken as reliable.
2. Gap set: data where the transaction fraud labels are unknown.
3. Test set: data of the day on which the algorithm is currently tested.

It corresponds therefore to a semi-supervised learning scheme (SSL), as training data are partially labeled. If the Gap set is totally left aside, this is an usual supervised learning (SL) problem again. Both cases (SL and SSL) were investigated:

- For the SL scheme, only the Training set is used to build the graph, and only the Training set is used to train the random forest.

- For the SSL scheme, the Training set and the Gap set are used to build the graph, and only the Training is used to train the random forest.

Once again, in arrangement with field experts, 15 days of training data and seven days for the gap set were chosen [5, 11]. This scenario is depicted on Fig. 1. Notice that on this figure, τ controls the testing day and that models are systematically built (overnight) on the 22 previous days. By changing τ , we get different testing days.

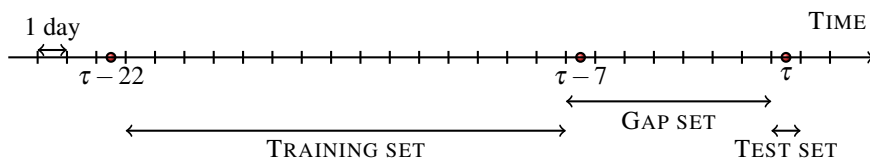


Fig. 1: Real-life FDS scenario with three sets of data. It takes several days to inspect all transactions, mainly because it is sometime the card holder who reports undetected frauds. Hence, in practice, the fraud tags of the Gap set are unknown. This scenario is repeated each day, as the parameter τ is incremented.

4.3 Including investigators feedback

Finally, even if in this last scenario it is not possible to know all fraud tags for the gap set, it is still conceivable that a fraction of previous alerts have been confirmed or overturned by human investigators (typically when a fraud alert occurs, the card is blocked and the card holder is contacted by phone). In our case, we put this number of feedbacks per day to 100, in arrangement with field experts. It is a realistic average number of cards than a human investigator can check per day, usually by contacting the card holder. So each day, the 100 most probable fraudulent card (according to the model) are checked and then used as feedback. So in each of our gap set (except in starting condition) 700 cards have been checked by human investigators. We will take advantage of these investigated cases in order to try to predict more accurately the fraudulent transactions. On average, it means that roughly 1400 transaction feedbacks (two transactions per card) from previous testing day (previous τ 's of our model) are available. This option will be referred as +FB and only make sense in a SSL scheme.

4.4 Removing merchant scores

Finally, we observed that removing merchant scores rises the performance. This is surprising at first glance but, after investigation, it turns out that new transactions involving new merchants cause issues (with our set-up, it corresponds to roughly 20% of merchants). In this case, the risk score is set to zero, causing the method to

under-evaluate the risk. This should clearly be tackled but we choose to let this for further work. This last option will be refers as noM.

5 Experimental comparisons

In this section, the possible variation of considered algorithms will be compared on a real-life e-commerce credit card transaction data set obtained from a large credit card issuer in Belgium. Those graph-based algorithms compute additional features and were presented in Section 3 and 4. For practical purposes, considered algorithms are recalled in Table 2 and the classifier is always a random forest with 400 trees.

The database is composed of 25,445,744 transactions divided in 139 days and fraud ratio is 0.31%. The features list can be found in Table 1. From this table, the first group contains socio-demographic features which are taken as-is. The second group contains the graph-based features described in Section 3 and 4. Notice that each transaction is linked with a card holder and with a merchant at a certain date: those three pieces of information (card holder, merchant and date) are used to build the tripartite graph. Finally, this database does not focus on a certain type of card fraud (stolen, card-not-present,...) but contains all reported fraudulent transactions in this time period.

Table 2: The nine compared models, see Sections 3 and 4 for acronyms. Considered variations of the APATE Algorithm according to four dimensions: merchant score status, hubs status, learning scheme and utilisation of feedback. Precision@100 (see Section 5) both for fraudulent card and transaction prediction is also reported (formatted mean \pm std)

Classifier name	Mer Score	Damp hubs	Learning	Feedback	Card Pr@100	Trx Pr@100
RWWR SL = APATE	used	no	Supervised	no	18.64 \pm 4.66	27.78 \pm 11.61
RWWR SSL	used	no	Semi-supervised	no	16.95 \pm 4.46	20.85 \pm 10.14
RWWR SSL +FB	used	no	Semi-supervised	yes	14.19 \pm 4.43	13.89 \pm 8.49
RCTK SL	used	yes	Supervised	no	23.78 \pm 9.52	40.50 \pm 18.00
RCTK SSL	used	yes	Semi-supervised	no	44.55 \pm 9.55	50.58 \pm 13.99
RCTK SSL +FB	used	yes	Semi-supervised	yes	37.15 \pm 10.14	49.06 \pm 14.70
RCTK noM SL	discarded	yes	Supervised	no	45.35 \pm 9.06	62.25 \pm 11.97
RCTK noM SSL	discarded	yes	Semi-supervised	no	56.08 \pm 8.06	81.61 \pm 9.00
RCTK noM SSL +FB	discarded	yes	Semi-supervised	yes	56.65 \pm 8.69	84.13 \pm 8.42

As a performance indicator, Precision@100 [27] was chosen, in accordance with field experts. It means that the 100 most probable (according to models) fraudulent transactions are checked by human investigators each day (and added as feedback in RWWR SSL +FB, RCTK w/ SSL +FB and RCTK noM w/ SSL +FB). Similarly all most probable fraudulent transactions are considered until 100 cards have been

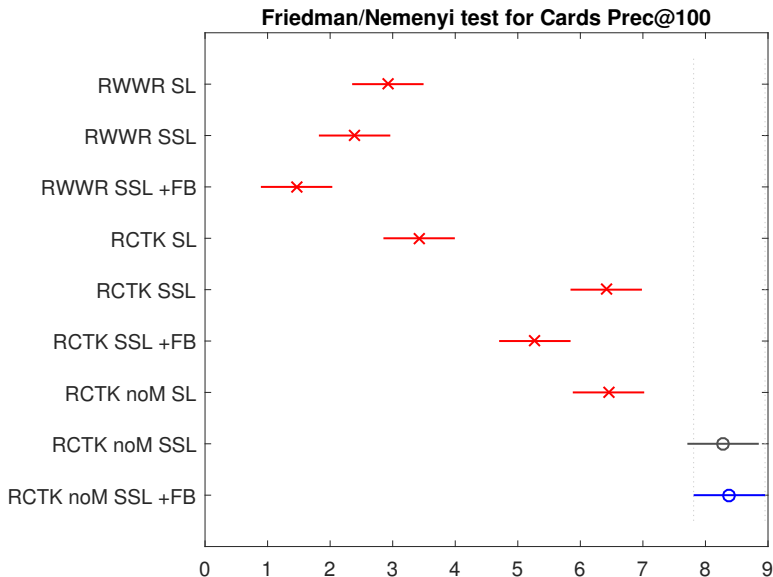


Fig. 2: Mean rank (circles and crosses) and critical difference (plain line) of the Friedman/Nemenyi test, obtained on a three-months real-life e-commerce credit card transaction data set. The blue (bottom circle) method has the best mean rank and is significantly better than red (crosses) methods. The Critical difference is 1.14. Performance metric is Pr@100 (Precision@100) on fraudulent card prediction.

checked as usually human investigators verify all transactions of a card when they investigate. Precision@100 reports the number of true fraudulent transaction or card among 100 investigated cards. Notice that this last metric is more realistic as it is somehow the normal work charge for a human investigators team.

Figure 2 compares methods from Table 2 through a Friedman/Nemenyi test [13]. To do so, we adopt a sliding window approach: each day (different τ from Fig 1) is considered as a different (train-gap-test) dataset. This test compares the ranking provided by Table 2 methods. Friedman null hypothesis is rejected with $\alpha = 0.05$ and Nemenyi critical difference is equal to 1.14. A method is considered as significantly better than another if its mean rank is larger by more than this amount.

Firstly, RCTK always beats RWWR, RWWR noM was therefore discarded. This superiority indicates that tackling the hubs problem is actually important.

Secondly, SSL leads to a huge improvement, but only if hubs have been damped. SSL predicted frauds tend to contain more frauds with a fraudulent activity during gap days, compared to SL ones. As the fraud tag is hidden for the gap set, it means that this information is obtained by network analysis (train+gap).

Thirdly, even if +FB bring some kind of information, it only increases performance when hubs are tackled (RCTK) and merchant scores are removed (noM). By the way, results are not significantly better on our three-months dataset. Further analysis

(not reported here) shows that with more data days and more checked cards, this improvement becomes significant (with $\alpha = 0.05$).

Lastly, removing merchant scores rises performance as explained in Section 4.

Overall, the best combination is RCTK noM SSL +FB, but it is not significantly better than RCTK noM SSL.

Finally, Figure 3 indicates the frequency of selected features by the random forest classifier. The method is RCTK SSL +FB and selects Mer scores most often. Sadly, new transactions involving new merchants cause issues. In this case, the risk score is set to zero, causing the method to under-evaluate the risk, resulting in a biased prediction. Discarding those four features (Mer scores) does increase the overall performance and selected variables of random forests stay similarly distributed.

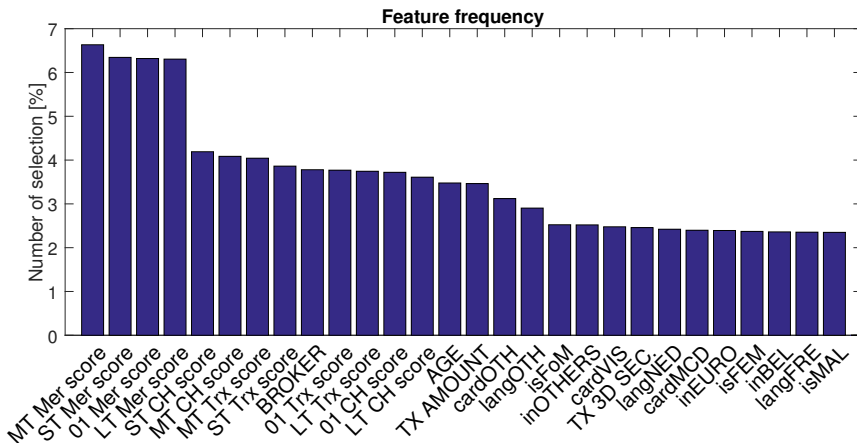


Fig. 3: Selected variables of random forests for the RCTK SSL +FB model for all days. Mer scores tend to bias the prediction. Discarding those four features does increase the overall performance (see Figure 2) and selected variables of random forests stay similarly distributed.

6 Conclusion

In this paper, we start from an existing Fraud Detection Systems (FDS) APATE and bring several improvements: which have a huge impact on performances damping hub nodes (RCTK), introduce restrictions due to real application (SSL, Gap set, Pr@100 as a metric) and introduce feedback information from human investigators (+FB). Those improvements multiply the Pr@100 by three, both on fraudulent card or transaction prediction (for acronyms, see Section 4).

However, introducing feedback does not lead to a significant improvement: feedback impact can be increased if more cards are checked, but this is non-realistic for investigators. New transactions involving new merchants are still an issue (see

noM in Section 4) which is left for further work: a possible way would be to mimic the learning procedure from [7]. Another envisaged further work is to introduce semi-supervised learning not only on graph analysis but also in main classifier.

Acknowledgements This work was partially supported by the Immediate and by the Brufence projects funded by Innoviris. We thank this institution for giving us the opportunity to conduct both fundamental and applied research.

References

- [1] Abdallah, A., Maarof, M.A., Zainal, A.: Fraud detection system. *Journal of Network and Computer Applications* **68**, 90–113 (2016)
- [2] Baesens, B., Van Vlasselaer, V., Verbeke, W.: *Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection*. Wiley Publishing (2015)
- [3] Bhattacharyya, S., Jha, S., Tharakunnel, K., Westland, J.C.: Data mining for credit card fraud: A comparative study. *Decision Support Systems* **50**(3), 602–613 (2011)
- [4] Bolton, R., Hand, D.: Statistical fraud detection: A review. *Statistical science* **17**, 235–249 (2002)
- [5] Bolton, R.J., Hand, D.J.: Unsupervised profiling methods for fraud detection. In: *Proceedings of the Credit Scoring and Credit Control VII Conference*, p. 235255 (2001)
- [6] Brandes, U., Erlebach, T.: *Network analysis: methodological foundations*. Springer-Verlag (2005)
- [7] Braun, F., Caelen, O., Smirnov, E., Kelk, S., Lebichot, B.: Improving card fraud detection through suspicious pattern discovery. Submitted for publication (2016)
- [8] of Certified Fraud Examiners, A.: Report to the nation (2002). URL http://www.acfe.com/uploadedFiles/ACFE_Website/Content/documents/2002RttN.pdf
- [9] Chapelle, O., Scholkopf, B., Zien, A.: *Semi-supervised learning*. MIT Press (2006)
- [10] Dal Pozzolo, A.: Adaptive machine learning for credit card fraud detection. Ph.D. thesis, Universite Libre de Bruxelles (2015)
- [11] Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., Bontempi, G.: Credit card fraud detection and concept-drift adaptation with delayed supervised information. In: *Proceedings of the International Joint Conference on Neural Networks*, pp. 1–8. IEEE (2015)
- [12] Dal Pozzolo, A., Caelen, O., Le Borgne, Y.A., Waterschoot, S., Bontempi, G.: Learned lessons in credit card fraud detection from a practitioner perspective. *Expert System with Applications* **10**(41), 4915–4928 (2014)
- [13] Demsar, J.: Statistical comparison of classifiers over multiple data sets. *Journal of Machine Learning Research* **7** pp. 1–30 (2006)
- [14] commerce Europe, E.: *Global b2c e-commerce light report 2015* (2014). URL <https://www.ecommerce-europe.eu/facts-figures/free-light-reports>
- [15] Fawcett, T., Provost, F.: Adaptive fraud detection. *Data Mining and Knowledge Discovery* **1**, 291–316 (1997)
- [16] Fouss, F., Francoise, K., Yen, L., Pirotte, A., Saerens, M.: An experimental investigation of kernels on a graph on collaborative recommendation and semisupervised classification. *Neural Networks* **31**, 53–72 (2012)
- [17] Hara, K., Suzuki, I., Shimbo, M., Kobayashi, K., Fukumizu, K., Radovanovic, M.: Localized centering: Reducing hubness in large-sample data. In: *Proceedings of the Association for the Advancement of Artificial Intelligence Conference*, pp. 2645–2651 (2015)
- [18] HSN Consultants, I.: The nilson report (2015). URL https://www.nilsonreport.com/publication_newsletter_archive_issue.php?issue=1068
- [19] Kemeny, J.G., Snell, J.L.: *Finite Markov Chains*. Springer-Verlag (1976)

- [20] Lebichot, B., Kivimaki, I., Françoise, K., Saerens, M.: Semi-supervised classification through the bag-of-paths group betweenness. *IEEE Transactions on Neural Networks and Learning Systems* **25**, 1173–1186 (2014)
- [21] Mantrach, A., van Zeebroeck, N., Francq, P., Shimbo, M., Bersini, H., Saerens, M.: Semi-supervised classification and betweenness computation on large, sparse, directed graphs. *Pattern Recognition* **44**(6), 1212 – 1224 (2011)
- [22] Newman, M.: *Networks: an introduction*. Oxford University Press (2010)
- [23] Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab (1999). Previous number = SIDL-WP-1999-0120
- [24] Phua, C., Lee, V., Smith-Miles, K., Gayler, R.: A comprehensive survey of data mining-based fraud detection research. *Computing Research Repository* **abs/1009.6119** (2010)
- [25] Radovanović, M., Nanopoulos, A., Ivanović, M.: Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research* **11**, 2487–2531 (2010)
- [26] Radovanović, M., Nanopoulos, A., Ivanović, M.: On the existence of obstinate results in vector space models. In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '10*, pp. 186–193. ACM (2010)
- [27] Theodoridis, S., Koutroumbas, K.: *Pattern recognition*, 4th ed. Academic Press (2009)
- [28] Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., Baesens, B.: Apate: A novel approach for automated credit card transaction fraud detection using network-based extensions. *Decision Support Systems* **75**, 38–48 (2015)
- [29] Weston, D.J., Hand, D.J., Adams, N.M., Whitrow, C., Juszczak, P.: Plastic card fraud detection using peer group analysis. *Advances in Data Analysis and Classification* **2**(1), 45–62 (2008)
- [30] Zaslavsky, V., Strizhak, A.: Credit card fraud detection using self-organizing maps. *Information and Security* **18**, 48 (2006)
- [31] Zhou, D., Bousquet, O., Lal, T., Weston, J., Scholkopf, B.: Learning with local and global consistency. In: *Proceedings of the Neural Information Processing Systems Conference (NIPS 2003)*, pp. 237–244 (2003)

Modeling City Locations as Complex Networks: An initial study

Lu Zhou, Yang Zhang, Jun Pang and Cheng-Te Li

Abstract Analyzing data collected from location-based social networks can reveal complex structure in human social relations. It can also lead to deep understandings of human mobility and help characterize city locations and their connectivity. In this paper, we construct location networks for six cities using a large-scale Instagram dataset. We find that these location networks share many topological features as in other different types of networks, along with properties specific to their cities. By mapping locations to their geographical coordinates, we further show that (1) our construction method can effectively reveal popular city locations, and (2) for two locations there is no clear correlation between their network distance and geographical distance. Moreover, all six location networks contain three or four large communities covering almost all locations in a city and the large communities in each city often exhibit clear spatial differences in geographical space.

1 Introduction

With the advancement of urbanization process, more and more people live in cities. The United Nation published a report in 2014 stating that 6 billion people will live in cities by 2050 (double the current amount). On one hand, city life brings people a lot of convenience. For instance, people can taste different cuisines and buy products from all over the world. On the other hand, it also results in many problems such as traffic jam and heat island effect. Many efforts have been taken to tackle these problems and improve people's life quality, such as the techniques of smart city. In particular, one fundamental component of cities, i.e., location, attracts a number

Lu Zhou · Yang Zhang · Jun Pang ✉

Faculty of Science, Technology and Communication & Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg, Luxembourg, e-mail: `firstname.lastname@uni.lu`

Cheng-Te Li (e-mail: `chengte@mail.ncku.edu.tw`)

Department of Statistics, National Cheng Kung University, Tainan, Taiwan

of attentions in academic research. Existing work exploited location information to infer users' friendship [16], recommend new locations [7] and measure urban deprivation [18]. However, most of these work treat locations separated from each other, while the interactions among locations are often not studied.

Online social networks have been the most successful application during the past decade, major companies including Facebook and Instagram have attracted a large number of users. With the development of mobile devices, social networks have been extended to geographical space. Nowadays, more and more social network users are sharing their photos or statues labeled with geographic information, namely *check-in*. The large-scale check-in data can be naturally used to describe human mobility, and provide us means to study location relations.

Network is one of the most common perspectives to study interactions in complex systems, it has attracted academia a particular interest in recent years, e.g., social networks, biological networks, transportation and computational history [17]. Locations can also be organized into networks, and such networks can be used to study connectivity among locations. Most of existing studies construct location networks based on user transitions among the locations [10, 13]. Namely, a location network is built as a directed weighted graph, whose nodes are locations and each edge is formed between two locations if a user directly moved between such two locations during a pre-defined time period. However, all these networks only reflect users' movement from one location directly to another in a short time, but they cannot describe the overall connections between locations. In our work, we present a different approach to construct location networks for six cities, including New York, Los Angeles, London, Paris, San Francisco and Tokyo, with more than 15 million check-in data collected from Instagram, and conduct empirical analyses on these networks to reveal their network features and properties related to location geographical coordinates. Our contributions can be summarized as follows:

- We propose a new method for constructing location networks (Section 2). In our construction, we treat each location as a node (similar to [10, 13]) and define a weighted edge between two nodes, which describes users' check-in behaviors and measures the strength of the connectivity between these two nodes.
- We adopt four measurements to describe the constructed location networks, including mean degree, degree distribution exponent, weighted clustering coefficient and average shortest path length. We find that location networks have similar topological features as in other complex networks. In addition, location networks also exhibit differences specific to their city (Section 3).
- We rank location popularity based on the PageRank algorithm [15], and show that our construction method leads to more effective rankings of locations when compared to rankings, e.g., based on location entropy [10] (Section 4).
- We reveal the relation between geographical distances and network distances between any two nodes (Section 4). To our surprise, we discover that there is no clear linear correlation between these distances.
- We detect network communities in our location networks, and discover that there only exist three or four large communities containing almost all the locations

Table 1: Summary of the original dataset

City	Users	Locations	Check-ins
New York	95,624	21,646	2,566,328
London	56,663	10,423	1,199,500
Paris	22,409	6,916	458,291
Los Angeles	85,788	19,412	2,055,290
Tokyo	35,487	19,610	835,896
San Francisco	25,374	7,302	585,727

in each city. Such communities also exhibit differences in geographical space (Section 5).

2 Network Construction

2.1 Dataset

Instagram is a photo-sharing social network service, with a fast growing user number. Instagram allows users to label locations when publishing photos. It is worth noticing that the location information on Instagram is imported from Foursquare, a social network that concentrates on location sharing. In addition, the authors of [11] have shown that Instagram users are much more willing to share locations than other social network users (e.g., 31 times more than Twitter users), which makes Instagram a suitable source for collecting check-in data.

To find users in the six cities, we start from querying Foursquare’s API to collect the location IDs in each city, together with each location’s rating. Then we use Instagram’s API to collect users’ check-ins in Instagram at the corresponding location IDs. In our experiment, we focus on users’ check-ins in year 2015. To resolve the data sparseness issue, for each city, we concentrate its users with at least 10 check-ins. Moreover, locations that are visited only by one user are filtered out. In the end, we use more than 7M check-ins in total to construct six location networks (one for each city). Table 1 summarizes our dataset.

2.2 Nodes, Edges and Weights

Our general goal is to study city locations from the network perspective, for instance, to understand why locations are connected to each other and how strongly the locations are connected. Along with a (complex) topological structure, many real networks display a large heterogeneity in the capacity and intensity of connections. Thus, it is important to have a measurement which reflects the relevancy between

any two locations in a city. Our intuition is to take city locations as nodes in the location network, locations are connected through users' check-in behaviors. In order to measure the strength of the connections, we need a measure to estimate and summarize how different users behave on the connections.

First, if two locations have been visited by one user, we consider that these two locations are associated with each other and there exists an edge between them. Let $L(u)$ denote the set of all locations that user u has visited. We define $E(u) = \{(\ell_i, \ell_j) \mid \ell_i, \ell_j \in L(u) \wedge \ell_i \neq \ell_j\}$ as the set of edges constructed from u 's check-in data. Based on this definition, we build an undirected graph, in which nodes denote (check-in) locations, an edges between any two nodes mean that the two locations have been visited by one user. Let V be the set of all check-in locations for a given city. We construct a location network $G = (V, E)$, where $E = \{(\ell_i, \ell_j) \mid (\ell_i, \ell_j) \subseteq V \times V\}$ captures all existing connections between two locations through users' check-in behaviors.

The next step is to quantify the strength for each edge (ℓ_i, ℓ_j) in G . An edge connecting two locations can be visited by many users, whose check-in behaviors can vary differently: some may visit many different locations while others visit only a few. Thus, we need to take users' active levels into account to measure edge weights. We adopt the *Shannon entropy* to quantify a user's active level (similar to the definition of location entropy to measure location's popularity [6]):

$$entropy(u) = - \sum_{\ell_i \in L(u)} p_{\ell_i} \cdot \ln(p_{\ell_i}) \quad (1)$$

where $p_{\ell_i} = u_{\ell_i} / \sum_{\ell_j \in L(u)} u_{\ell_j}$ describes the probability that user u visited location ℓ_i . It is easy to see that $entropy(u)$ depends on both the diversity of locations and the frequency how u visited those different locations.

Then, we use users' active levels to define two locations' (ℓ_i and ℓ_j) edge weight as the following:

$$W_{\ell_i \ell_j} = \sum_{u \in U_{\ell_i \ell_j}} entropy(u) \cdot \frac{\sqrt{u_{\ell_i} u_{\ell_j}}}{\sum_{(\ell_s, \ell_t) \in E(u)} \sqrt{u_{\ell_s} u_{\ell_t}}} \quad (2)$$

where $U_{\ell_i \ell_j}$ represents the set of users who visited both locations ℓ_i and ℓ_j , u_{ℓ_i} denotes user u 's number of check-ins at ℓ_i . The factor $\sqrt{u_{\ell_i} u_{\ell_j}}$ reflects how often u visits both ℓ_i and ℓ_j , and it is normalized by all pairs of locations u has visited. By multiplying $entropy(u)$ and $\frac{\sqrt{u_{\ell_i} u_{\ell_j}}}{\sum_{(\ell_s, \ell_t) \in E(u)} \sqrt{u_{\ell_s} u_{\ell_t}}}$, it shows how much a user u 's check-in behavior can be cast onto the edge (ℓ_i, ℓ_j) . Furthermore, in order to reflect the contributions made by all the users who visited both ℓ_i and ℓ_j , we sum up each user's contribution to have the final weight for the edge (ℓ_i, ℓ_j) .

In the end, we construct a weighted graph, representing a location network for each city: $G = (V, E)$, with a function $W: E \rightarrow R^+$ assigning a positive value to every edge in G . For each $(\ell_i, \ell_j) \in E$, we have $W(\ell_i, \ell_j) = W_{\ell_i \ell_j}$ as defined in Equation 2. Table 2 summarizes the numbers of nodes and edges, the minimal, maximal and mean

Table 2: Statistics for the six constructed location networks

City	Nodes	Edges	Min Weight	Max Weight	Mean Weight
New York	21,646	5,697,507	8.9673e-05	286.2492	0.0383
London	10,423	1,861,304	2.1687e-04	165.5303	0.0647
Paris	6,916	655,793	1.7152e-04	655.1800	0.0665
Los Angeles	19,412	3,881,191	8.5379e-05	381.1223	0.0468
Tokyo	19,610	2,635,335	1.0492e-04	157.8897	0.0287
San Francisco	7,302	1,334,240	1.7223e-04	100.6024	0.0407

weights, for the six constructed location networks. For all the location networks, most edges have small weights (this can be concluded from the mean weights), while, there exist a few edges with large weights, which, we suppose, often connect the most popular locations in the cities. In order to confirm our hypothesis, we obtain each pair of locations which are connected by an edge with the largest weight for each city. For example, there exists an edge with the largest weight connecting Rockefeller Centre and Times Square, which are known as two of the most famous locations in New York City. Similarly, the other location pairs are also the most famous places in the corresponding cities: Tower Bridge and Tower of London in London, Musee du Louvre and Notre-Dame de Paris in Paris, Staples Centre and Dodger Stadium in Los Angeles, Tokyo Disneyland and Tokyo Dome in Tokyo, AT&T Park and Union Square in San Francisco.

In addition, it is easy to see that New York city has the largest location network, followed by Los Angeles and Tokyo, and then by London. Paris and San Francisco have relatively small location networks. The reason for the different sizes of location networks are mostly due to the difference between the population and the size of each city. We list all the basic information (i.e., population, size and density) of these six cities in Table 3. From Tables 1, 2 and 3, we find that New York has the largest population density, and its number of Instagram users, check-ins, nodes and edges are also the highest. Meanwhile, all the corresponding parameters in Paris are the smallest. Thus, we observe a positive correlation between the size of our location networks and city's population density.

3 Graph Measurements and Analysis

After constructing the (weighted) location networks, we analyze their properties to have a representative description of these networks. In the following, we present four measurements, as discussed in [12].

Mean degree (MD). Even though our network is weighted, to compare with other networks, we consider the unweighted mean degree here. That is, for each node, we

Table 3: Basic information for each city

City	Population	Size	Population density
New York	8.41	789	10,659.06
London	8.67	1,572	5,515.27
Paris	12.29	12,012	1,023.14
Los Angeles	9.82	1,214	8,088.96
Tokyo	13.62	2,188	6,224.86
San Francisco	0.84	121	6,942.15

compute its number of degrees and calculate the mean of all the nodes' number of degrees as mean degree.

Degree distribution exponent (DDE). In general, the degree distribution is the probability distribution of these degrees over the whole network. For our location networks, we need to construct a histogram of the degrees. Like other power-law degree distributions, the histogram is highly right-skewed. It means that its degree distribution has a long right tail of values that are far above the average, indicating that more nodes have smaller degrees while less ones have much larger degrees. To describe the degree distribution of each location network, we compute the exponent α for each degree distribution curve: $p_k \sim k^{-\alpha}$, where p_k denotes the number of each degree k , and α means the exponent for each degree distribution curve.

Weighted clustering coefficient (WCC). Clustering coefficient captures the degree to which nodes in a graph tend to cluster together. In other words, it is related to the number of closed triangles in the neighborhood of a node. Here, we apply the local clustering coefficient algorithm [14, 19]. For a node, its clustering coefficient is the fraction of the number of present links over the total number of possible links between its neighbors. Therefore, the outcome strictly ranges between 0 and 1, where 0 denotes that no links exist between the neighbors, and 1 if all possible links exist. The equation for clustering coefficient of any node in a location network is given as

$$WCC_{\ell_i} = \frac{\sum_{\ell_j, \ell_k \in Cnei(\ell_i)} W_{\ell_i, \ell_j} + W_{\ell_i, \ell_k}}{\sum_{\ell_m, \ell_n \in Nei(\ell_i) \wedge \ell_m \neq \ell_n} W_{\ell_i, \ell_m} + W_{\ell_i, \ell_n}} \quad (3)$$

where $Cnei(\ell_i)$ denotes a set of pairs of locations which are both neighbors of ℓ_i and are also connected in the network, namely closed triplets. For each node, we sum the value of the closed triplets that are centred on the node and divide it by the total value of all triplets centred on the node. The larger coefficient of one location implies that user who visits this location will also visit its neighbors more frequently.

After obtaining the coefficient for each location, we compute the average clustering coefficient value for the whole network. To some extent, the average value reflects the density of the whole network.

Table 4: Properties comparison with six cities

City	Nodes	Edges	MD	DDE α	WCC	CRC
New York	21,646	5,697,507	526.426	3.818	0.011	-9.5e-4
London	10,423	1,861,304	357.153	3.116	0.013	0.0448
Paris	6,916	655,793	189.645	2.185	0.019	0.0670
Los Angeles	19,412	3,881,191	399.875	3.500	0.014	0.0560
Tokyo	19,610	2,635,335	268.775	3.493	0.014	0.0427
San Francisco	7,302	1,334,240	365.445	3.970	0.012	0.0339

Analysis. Table 4 summarizes the computed four measurements for the six location networks. We compare their features with other typical real-life networks, such as biological networks and technological networks. First, we measure the proportion of existed edges to all possible edges in each location network (New York: 0.024, London: 0.034, Paris: 0.027, Los Angeles: 0.021, Tokyo: 0.014, San Francisco: 0.050). Compared with the given statistics of a number of published networks [12], we can find that the edge proportions in our location networks are higher than other networks, where most edge proportions are less than 10^{-4} . This implies that most locations in our networks have more connections with others. Furthermore, the nodes with more connecting edges are often popular locations in the cities. Our location networks' mean degrees are much bigger than other networks [12] (e.g., WWW Altavista network: 10.46, physics coauthorship: 9.27, metabolic network: 9.64), which is mostly due to the large numbers of edges in our location networks. On the other hand, clustering coefficient values of our location networks are much lower than other networks [14]. The main reason for this result is that, due to Equation 3, the denominator can be influenced largely by the number of edges in networks. As our networks have a plenty of edges, the values of clustering coefficient arrive at a low level ¹. Since the clustering coefficient measures the density of triangles in a network, higher value means the networks are much denser and the neighbors of a node in the network are more likely to be connected. The computed clustering coefficients in Table 4 imply that the neighbors of one location are not necessarily connected and influenced by other locations. When considering the exponent α , we find that our network and other networks have similar values. It means the degree distribution of our location networks also follows a power law form. In general, we can conclude that location networks share basic topological features of complex networks while having their own characteristics.

Table 4 also shows differences among the six cities. London's location network has the largest mean degree, even though the location network of New York has a much larger number of edges than London. This is due to that the edge proportion in New York's location network is much lower than London's, and its degree distribution

¹ Clustering coefficient for unweighted graphs will increase when there are more edges. However, WCC considers not only the number of edges, but also the edge weights. Since most of the edges in our location networks have small weights, WCC will decrease when there are more edges.

curve also has a longer tail than London's. This is also reflected in the clustering coefficients: London's location network has a higher clustering coefficient than New York's, meaning that its edge weights are relatively larger. For another example, Tokyo has many more locations than Paris, and the numbers of nodes and edges in its location network are much larger than Paris' network. However, the clustering coefficient in Tokyo's location network is smaller than Paris'. Meanwhile, Tokyo's location network's exponent α is higher than Paris', which means most nodes in Tokyo's network have much smaller degrees than nodes in Paris' location network. These also explain why the mean degree of Tokyo's location network (i.e., 7.717) is much smaller than Paris's (i.e., 12.612). We also observe that the shortest path length of location network in Paris is the largest while New York has the smallest shortest path length. From these, we can conclude that each location network has features specific to their city.

4 Location Ranking and Distances

4.1 Location Ranking

Location popularity has received many attentions in recent years, it is considered as an essential part for building real-world applications such as location recommendation and friendship prediction. With our location networks constructed, we want to measure each location's popularity as well. To do so, we adopt one of the most classical algorithms on measuring the popularity of nodes in networks, i.e., PageRank.

To further validate our popularity measurement, we take location entropy [6] for comparison. Location entropy is one common measurement for location popularity. If a location has a high entropy, it shows that many different users have visited the location, thus the location is popular. To compare popularity ranking on locations, based on the results of executing PageRank on our networks and computing the location entropy of each node in our networks, we adopt a quantitative way with the help of location ratings. As mentioned in Section 2, due to the connection between Foursquare and Instagram's APIs, when collecting check-in data at a certain location, we are able to get the location's rating from Foursquare (between 1 and 10). We treat these ratings as a ground truth to rank location popularity.

We compute the correlation coefficients between ratings and both PageRank scores and location entropies for all locations in each city. The results in Table 5 show that PageRank scores are much more correlated with ratings than location entropies. With the assumption that high ratings indicate popular locations, we conclude that PageRank scores, computed on our location networks, are very effective in evaluating location popularity. This further demonstrates the usefulness of our network construction method (note that location entropy only concentrates on the check-ins of each location, but not on the relations among locations).

Table 5: Comparison between correlation coefficients obtained by PageRank and location entropy

City	PageRank location entropy		City	PageRank location entropy	
New York	0.4356	0.1420	Los Angeles	0.4714	0.1885
London	0.3774	0.2039	Tokyo	0.4169	0.1720
Paris	0.3952	0.1482	San Francisco	0.4449	0.2079

4.2 Location Distances

Besides location popularity, locations have other important properties, i.e., their geographical coordinates. Where a location is geographically located, in many cases, can determine its fundamental properties such as its functionalities and its value in the real estate market. In this subsection, we study the relation between locations organized as a network and their actual geographical coordinates.

To our knowledge, human mobility is probably constrained geographically by the distance one can travel within a day [9]. Thus, we assume that the geographic distance (Geo_{dist}) between any two locations has an influence on the connectivity between them in location networks. In other words, Geo_{dist} has an impact on user's judgement on the choice of visiting different locations. Thus, we need to obtain the check-in correlation coefficient between Geo_{dist} and Net_{dist} (i.e., network distance, that is, the shortest path length in the network between any two locations), in order to determine whether there exists any correlation between such two kinds of distance. In particular, Geo_{dist} means the actual geodesic distance computed by the Euclidean distance using latitude and longitude values, while the weights on edges in the constructed location network can be considered as Net_{dist} . Finally, we get the correlation coefficient between these two kinds of distance for each city, shown in last column (CRC) in Table 4, where most of the correlation coefficient values are close to zero. This suggests that there is no linear relation between geographical distance and edge weight defined in the paper.

5 Community Analysis

To further understand the structure of our location networks, we perform analysis on their community structures. In simple terms, a community is a subset of nodes in a network with links among the community members are much more than between the community and the rest of the network. According to [20], the community structure is one of the most useful granularity to study networks, it has been used by researchers, e.g., to study interactions between modules [1] and predict unobserved connections [4].

Table 6: Summary of detected largest communities for each city

City	Comm. 1	Comm. 2	Comm. 3	Comm. 4	#. Comm.	% Top communities
New York	7,679	5,775	5,678	2,473	18	99.8%
London	3,626	3,450	3,299	-	15	99.5%
Paris	2,775	2,333	1,646	-	11	97.7%
Los Angeles	6,309	5,326	4,611	3,133	14	99.8%
Tokyo	6,865	5,845	4,884	1,918	21	99.5%
San Francisco	2,524	2,238	2,192	-	16	95.2%

5.1 Network Community

We adopt one of the classic approaches, namely the fast greedy modularity optimization algorithm [5] (fast greedy), to detect network communities in our location networks. The algorithm is essentially a fast implementation of the first community detection algorithm based on modularity optimization [8]. Starting from a set of isolated nodes, the fast greedy algorithm adds edges from the original graph to maximize the modularity [5] of the newly generated graph at each step.

We obtain multiple network communities for each city's location network with many communities only containing a few locations while several large (three or four) ones containing most of the locations in the network. Table 6 presents the statistics on the sizes of the largest communities in each city. For instance, the top 4 communities in New York and Los Angeles contain more than 99.8% of the locations in these two cities. This observation corresponds well to other complex networks [5] for containing only a few large components in their network structures.

5.2 Geographical Community

Next, we project network communities into a geographical space, and find that different communities are associated with different geographical signatures. Figure 1 exhibits the network communities in New York. The locations in the black community in New York (top left in Figure 1) are located throughout the city, including Manhattan, Brooklyn and Queens, while the blue and green communities mainly concentrates on Manhattan. Moreover, most locations of the blue community are in midtown, downtown and upwest while the green community has many locations in upeast side and the central park. Meanwhile, the locations in the red community are distributed more uniformly compared to other three communities with an interesting concentration in Jersey city. Similar observations can be made in all other five cities.

Recall the conclusion in Section 4.2 that there is no obvious correlation between geographical distances and edge weights. Meanwhile, in this section we find that different communities (partitioned based on the densely connected edges) distribute at different geographical spaces. This seems to be a contradiction. However, we

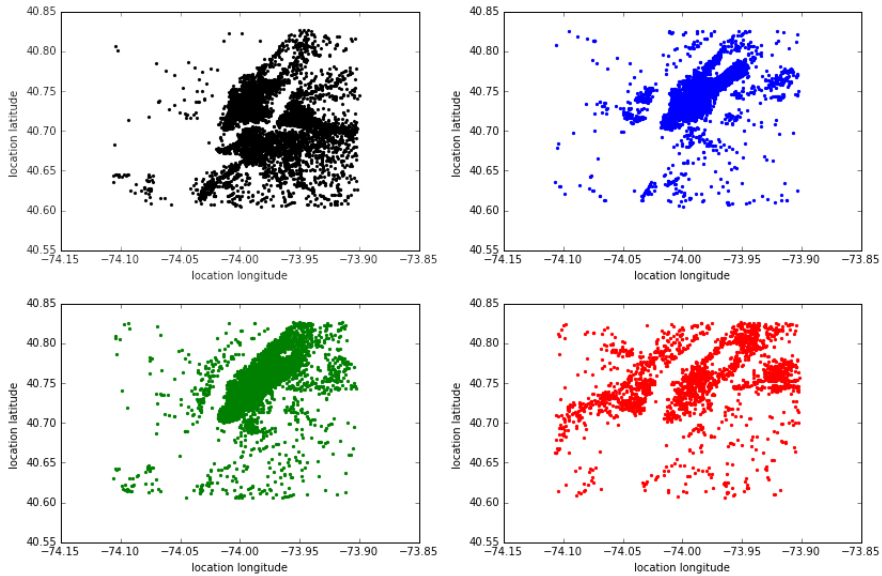


Fig. 1: The largest communities of New York shown on the map separately

need to mention that locations of different communities are not located completely differently, and most communities are overlapped in city centers.

6 Conclusion and Future Work

In this paper, we have constructed weighted location networks based on check-in behaviors of millions of Instagram users across six cities in the world. Our initial study of the constructed location networks has focused on their basic features as defined for complex networks. Moreover, we mapped locations in each city to their corresponding geographical coordinates, and discovered that our construction method is effective in revealing popular locations. We also discovered that there is no linear correlation between geographical distance and our edge weight. For each location network, we found a few largest communities covering almost all locations in a city, as well as such communities have an obvious distribution geographically.

The way how we constructed location networks has an emphasis on quantifying a user’s over-all check-in behavior and then distributing it to all edges connecting two locations that the user has visited. It is interesting to compare our construction methods with other methods for constructing location networks from location check-ins. Our next step is to further analyze community structure of each location network, e.g., through the use of different community detection algorithms and the study of their location category distributions. For instance, we want to apply the recently proposed clustering method in [3] to check whether location networks also consist a higher-order organization. Based on a larger Instagram dataset we have collected, we

will also investigate growth models for location networks, for example, by following the application of preferential attachment to the growth of the Internet [2].

References

- [1] Airoldi, E.M., Blei, D.M., Fienberg, S.E., Xing, E.P.: Mixed membership stochastic block-models. *Journal of Machine Learning Research* **9**, 1981–2014 (2008)
- [2] Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. *Science* **286**(5439), 509–512 (1999)
- [3] Benson, A., Gleich, D., Leskovec, J.: Higher-order organization of complex networks. *Science* **353**(6295), 163–166 (2016)
- [4] Chang, J., Blei, D.M.: Relational topic models for document networks. In: Proc. 12th International Conference on Artificial Intelligence and Statistics (AISTATS), *JMLR Proceedings*, vol. 5, pp. 81–88. JMLR.org (2009)
- [5] Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. *Physical Review E* **70**(6), 066,111 (2004)
- [6] Cranshaw, J., Toch, E., Hone, J., Kittur, A., Sadeh, N.: Bridging the gap between physical location and online social networks. In: Proc. 12th ACM Conference on Ubiquitous Computing (UbiComp), pp. 119–128. ACM (2010)
- [7] Gao, H., Tang, J., Hu, X., Liu, H.: Exploring temporal effects for location recommendation on location-based social networks. In: Proc. 7th ACM Conference on Recommender Systems (RecSys), pp. 93–100. ACM (2013)
- [8] Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* **99**(12), 7821–7826 (2002)
- [9] Gonzalez, M.C., Hidalgo, C.A., Barabasi, A.L.: Understanding individual human mobility patterns. *Nature* **453**(7196), 779–782 (2008)
- [10] Hristova, D., Williams, M.J., Musolesi, M., Panzarasa, P., Mascolo, C.: Measuring urban social diversity using interconnected geo-social networks. In: Proc. 25th International Conference on World Wide Web (WWW), pp. 21–30. ACM (2016)
- [11] Manikonda, L., Hu, Y., Kambhampati, S.: Analyzing user activities, demographics, social network structure and user-generated content on Instagram. *CoRR* **abs/1410.8099** (2014)
- [12] Newman, M.E.J.: The structure and function of complex networks. *SIAM Review* **45**(2), 167–256 (2003)
- [13] Noulas, A., Shaw, B., Lambiotte, R., Mascolo, C.: Topological properties and temporal dynamics of place networks in urban environments. In: Proc. 24th International Conference on World Wide Web (WWW Companion), pp. 431–441. ACM (2015)
- [14] Opsahl, T., Panzarasa, P.: Clustering in weighted networks. *Social Networks* **31**(2), 155–163 (2009)
- [15] Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab (1999)
- [16] Scellato, S., Noulas, A., Mascolo, C.: Exploiting place features in link prediction on location-based social networks. In: Proc. 17th ACM Conference on Knowledge Discovery and Data Mining (KDD), pp. 1046–1054. ACM (2011)
- [17] Schich, M., Song, C., Ahn, Y.Y., Mirsky, A., Martino, M., Barabasi, A.L., Helbing, D.: A network framework of cultural history. *Science* **345**(6196), 558–562 (2014)
- [18] Venerandi, A., Quattrone, G., Capra, L., Quercia, D., Saez-Trumper, D.: Measuring urban deprivation from user generated content. In: Proc. 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW), pp. 254–264. ACM (2015)

- [19] Watts, D.J., Strogatz, S.H.: Collective dynamics of ‘small-world’ networks. *Nature* **393**(6684), 440–442 (1998)
- [20] Yang, J., McAuley, J.J., Leskovec, J.: Detecting cohesive and 2-mode communities in directed and undirected networks. In: Proc. 7th ACM International Conference on Web Search and Data Mining (WSDM), pp. 323–332. ACM (2014)

An analysis of the Bitcoin users graph: inferring unusual behaviours

Damiano Di Francesco Maesa, Andrea Marino and Laura Ricci

Abstract An increasing interest on cryptocurrencies has recently raised, in particular on bitcoin. A unique feature of this system is that the list of all the economic transactions is publicly available. This makes available a large amount of information that can be analysed to discover the topological properties of the transaction graph and to obtain insights in the behaviour of the users. In a previous work we have presented a first set of analyses of the bitcoin network. Among other properties of the network, these analyses have also revealed a set of unusual patterns in the bitcoin users graph. We conjecture that these topological patterns are due to artificial users behaviors, not strictly related to normal economic interaction. In particular, in this paper, we analyse the outliers in the in-degree distribution of the bitcoin users graph. The results of our analysis support our conjecture, i.e. they are due to artificial transaction patterns.

1 Introduction

The boost in the diffusion, during the last years, of bitcoin [12], the first true digital currency, together with the public availability of its blockchain makes it interesting and feasible to analyse the behaviour of the users of this peculiar economy. Even if bitcoin still represents a niche economy, it is no longer an experimental currency only for computer science specialists, and has reached a widespread usage. Therefore, the analysis of its blockchain may return interesting insights on the behaviour of the users of a cryptocurrency.

Our previous work [7] analysed several properties of the bitcoin users graph. In particular, we showed that the graph presents many features characteristics of the small-world phenomenon, but also some odd behaviours. As a matter of fact, while

Damiano Di Francesco Maesa (e-mail: damiano.difrancescomaesa@for.unipi.it) ·
Andrea Marino (e-mail: andrea.marino@unipi.it) · Laura Ricci (e-mail: laura.ricci@unipi.it)

Department of Computer Science, University of Pisa

the average distance between nodes is low, the graph presents a high value of the diameter. This highlights the presence of a few pair of nodes connected by long paths. Furthermore, the in-degree distribution of the nodes presents some relevant outliers. A possible conjecture is that these odd behaviors are caused by users exploiting bitcoin not only for ordinary transactions, but rather for other activities, like fund management and, possibly, attacks. In other words, our conjecture is that if we could distinguish and isolate the transactions representing ordinary economic interactions from the other ones, the resulting user graph would be a better representation of a small world.

This paper will focus on the analysis of the outliers present in the in-degree distribution of the users graph. Our analysis shows that the outliers are a consequence of particular anomalous chains of transactions, that we classify as PS-transactions (Pseudo-Spam transactions). We give different conjectures about the meaning of these transactions and analyze their features.

We are currently also investigating the anomalous network diameter and our preliminary results show that its unexpected high length is caused by the behaviour of a single user. So those chains can be further agglomerated in just one cluster significantly lowering the diameter length and hence obtaining a much shorter diameter as expected in a small world. However, due to space constraints, these results are not shown in this paper.

The paper is organized as follows: in Section 2 we report some related work. Section 3 presents our analysis inferring unusual behaviors of the users of the bitcoin network and our conjectures about such behaviors. Section 4, presents a refinement of our initial analysis which is exploited in Section 5 to prove our initial conjecture. Finally, Section 6 reports our conclusions.

2 Related Work

Several features of the bitcoin network have been recently analysed. Most analyses are based on a "transaction graph" built from the blockchain, which is, in turn, transformed in the "users graph" (multi-graph with sets of addresses as nodes and arcs derived from the transaction graph) through a well established heuristic rule. By applying this rule, all the input addresses of a multi-input transaction are considered as belonging to the same user [8, 12] (and we say that they are *clustered* in a *cluster* representing the user). This heuristic rule, possibly combined with other heuristic rules, has been used for several analyses [4, 9, 10, 11, 13].

Our previous work [7], has highlighted several properties of the bitcoin network, detected by studying the time evolution of the network in the last years. We have observed a small average distance and we have characterized the network as a small-world. Moreover, we have computed also the diameter by using the algorithm in [6]. Surprisingly, we have observed an high diameter and the presence of several outliers in the in-degree distribution of the nodes. We found that the most central nodes in the network (according to harmonic centrality [5]) are also the ones with highest degree. Finally, we verified the rich get richer conjecture, both from the point of view

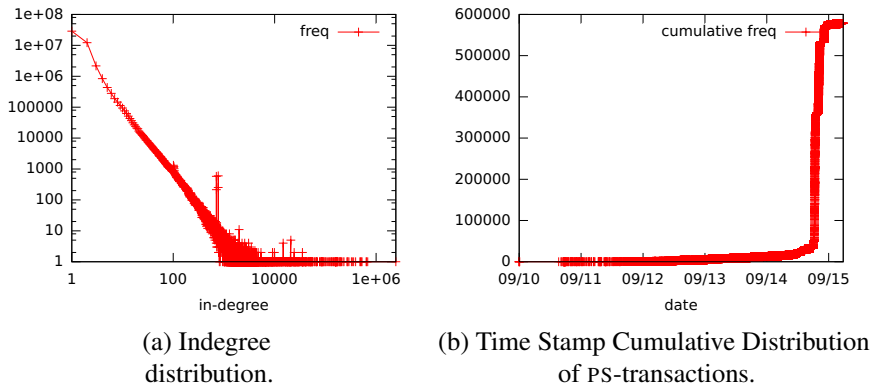


Fig. 1

of the balance of each node and from the connectivity point of view. The analyses we conducted revealed the presence of unusual patterns in the bitcoin graph. The goal of the following sections is to study one of these anomalous patterns.

3 Analyzing Indegree Outliers and Detecting PS-transactions

As observed in [7], the indegree distribution of the nodes of the users graph follows a power law, as expected in a small world network. For the sake of completeness we report this in indegree distribution in Figure 1(a) (note the log-log scale). In [7] we also observed that the exponent of this distribution is stable over time. (i.e. the exponent for the distribution of the indegrees of the graph obtained from the blockchain at discrete time intervals). In this paper, we aim to verify the following conjecture.

Conjecture 3.1. The small world theory discrepancies, as the indegree distribution outliers, are caused by artificial users behavior.

Restricting ourselves to the indegree distribution analysis only, we want now to verify the hypothesis by proving that such distribution outliers are in fact a consequence of long chains of special transactions.

In order to select the indegree outliers to be analyzed, we consider all the degrees having the following property (with 10 as value of the parameter k).

Property 3.1. Let $y = I(x)$ be the indegree distribution, where $I(x)$ is the number of nodes of the users graph with indegree x , and let Δ be the maximum indegree in the graph and $k \in \mathbb{N}$ a parameter. For every $k < x \leq \Delta$, x is a *suspicious outlier* if $I(x)$ is at least one order of magnitude greater than the average $I(x-i)$ with $1 \leq i \leq k$, i.e. if
$$I(x) > 10 \cdot \frac{\sum_{i=1}^k I(x-i)}{k}.$$

The only spikes satisfying the Property 3.1 are the ones corresponding to indegree 708, 709, 771, and 772. The corresponding $I(x)$ values are between two and three

order of magnitude above the value expected. We have analyzed the nodes of the network corresponding to these peculiar degrees.

To analyze the selected outliers, we started from a manual inspection of them. To do so we retrieved in the graph the clusters with indegree 708, 709, 771 and 772 and isolated their neighborhood and transactions history. This provided us with 1647 clusters to analyze. We noticed that many of them had almost consecutive identifiers in our graph. That happened because the oldest addresses contained in each of those clusters appeared for the first time in the blockchain together as destination of a payment in a unique transaction.

Looking for the first appearance of a sample of those addresses in the blockchain we noticed some peculiar transactions. One of these, for instance (Transaction hash 35dead89c059e846e2013a06a70cd84a7ba0f80da7741c283d6efd573e0a7319) has one input and 101 outputs paying 0.00001 BTC to each one of the outputs except one, that is filled with the change (minus the fees). The address containing the change is then used to perform an analogous transaction leaving the change in a new address and so on. Basically the behavior of the transaction creator is to create a chain of transactions, where a transaction at each step pays a constant amount of 0.00001 BTC to some addresses and leaves the change in an intermediary address used as input for the next hop in the chain. A chain ends either when the funds in the last change address are used for a transaction without this particular structure or when the input funds are completely spent and no change address is used in the last transaction of the chain. We also noted that the output addresses receiving 0.00001 BTC were addresses for the most part identifiable with users from the `bitcointalk` forum (indeed, the forum users had specified those addresses in their signatures). This suspicious transactions chain led us to define a new classification for transactions, labeling all the transactions with this peculiar behavior as *pseudo-spam* transactions (PS-transactions).

Definition 3.1. Let A be the set of all addresses present in the blockchain. Given a transaction t modeled as a tuple $(In, Out, InAmount, Fees)$, where:

- $In \subseteq A$;
- Out is a multiset of couples (o, b) where $o \in A$ and $b \in \mathbb{R}$, where b corresponds to the amount paid to address o by In ;
- $InAmount \in \mathbb{R}$;
- $Fees \in \mathbb{R}$;

we say that t is a *pseudo-spam transaction* (PS-transaction) if it satisfies the following properties:

- $|In| = 1$;
- $|Out| \geq 2$;
- $|\{(o, b) \in Out : b \neq 0.00001\}| \leq 1$.

In other words, a PS-transaction is such that the only input address pays all the others (at least two) 0.00001 BTC except one which can be the recipient of an arbitrary amount. We call this particular address *change address*, i.e. a is a change

address if a is s.t. $(a, b) \in Out_i$ and $b \neq 0.00001$. We call *common output* any output containing an address that is not a change address. Of course in each PS-transaction can exist at most one change address.

3.1 On the Economical Meaning of PS-transactions

Artificial transactions are not uncommon in the blockchain. Since they target what, at first glance, seems like a random selection of addresses in the blockchain, some users in the past have noticed receiving unexpected payments from such transactions and some interest has sparked around them. Unfortunately, there is no clear explanation of the goal of such transactions. In particular, in the following we explore some possible existing conjecture, showing that none of them is able to fully explain the purpose of our PS-transactions.

It is possible that these transactions are part of an attack on users pseudonymity, as an attempt to link addresses ownership. In fact the amounts sent are so tiny that in order to be spent they must be first combined with other funds in a multi-input transaction. This would potentially reveal new linking for the multi-input clustering heuristic (see Section 2) increasing its effectiveness. Even if this theory sounds reasonable, it does not seem to be applicable to our observed real use case. In fact the targets of our manually observed suspicious transactions are not picked at random from the blockchain but derived from a very close set of users belonging to the `bitcontalk` forum, and the transactions pay the same amount to any address multiple times. For an attacker it would make little sense to send funds (hence spending them) to the same address a lot of times and would be more efficient to send those funds to different addresses instead, because this would increase its probability of triggering a funds consolidation while minimizing the cost of the attack.

Another possible conjecture is that those transactions are used as part of a spam attack, to fill the blockchain space with useless data. But this is arguably not true since most of those transactions pay a regular fair fee to be included in the blockchain and so they have the same right to be included as any other transaction. Note that we perform a transaction analysis based on the blockchain information, so we only consider the permanent effect of transactions. The kind of transaction observed can be effectively used to perform a live spam attack to rapidly fill the users pending transactions lists, as historically really happened during the flooding attack of July 2015 [1]. But live spam attacks by themselves leave little to no sign on the blockchain.

Another possible interpretation would be that this transactions are used for advertising. By using vanity addresses or inserting human readable messages in the transactions it is possible to use a transaction to cheaply save an advertisement message in the blockchain forever. By including in such transactions the largest possible number of outputs one may attempt to increase the message visibility.

A famous example of these transactions arose to popularity during the Sochi Olympics, because two addresses (`1SochiWwFFySPjQoi2biVftXn8NRPCSQC` and `1Enjoy1C4bYBr3tN4sMKxvvJDqG8NkdR4Z`) started sending thousands of transactions paying exactly 1 satoshi (0.00000001 BTC) to what seemed like

random addresses read directly from the blockchain. Those transactions payed no fees and so only few of them were actually saved in the blockchain but they remained for hours in the users wallets as unconfirmed transactions, gaining a lot of visibility [2, 3]. It seems difficult to think that this was part of a deanonymization attack since most of the transactions never became part of the blockchain and so could not be spent to possibly reveal addresses linking. It might have been considered a spam attack but only limited to the live network (by filling the unconfirmed transaction lists of the users with useless data) but it had very little effect on the blockchain since few transactions were actually included. So the most plausible theory seems to be that it was part of a temporary spam advertising campaign, and a successful one since most bitcoin users received the message to “Enjoy Sochi” with very little cost. The cost was so little since very few transactions were accepted in a block (hence actually spending the used funds) and the 0.00000001 BTC payments carried so little value to do not matter anyway.

Whatever is the reason for this kind of transactions, it is obvious that they should be considered artificial transactions anyway, since the transaction purpose is to obtain some kind of side real world effect rather than to transfer value between addresses. This is clearly obvious for the Sochi example where the fair fees cost of a transaction would exceed the value effectively transferred. The same can be said for our manual inspected transactions where the fee was 0.0007184 BTC, hence seventy times the single amounts transferred and 41.8% of the total value actually spent by the transaction. This is the reason why we have labeled this kind of transactions as “pseudo-spam” even if we do not know neither want to imply that they are part of a spam attempt.

3.2 Chaining PS-transactions

Applying Definition 3.1 to our dataset we labeled 578 316 transactions as PS-transaction, out of the 99 602 440 multi-input multi-output transactions contained in our database. The transactions vary a lot (considering most of the transactions features as the number of outputs or the fees payed), but an interesting behavior can be seen analyzing the timestamps cumulative distribution among those transactions. As shown in Figure 1(b) we can see a steep step during July 2015 showing that most of those transactions were performed at that time. This is consistent with our observations, since the transactions of our case study take place during July 2015 as well, and with the existence of an historically recorded flooding attack happened during the same period [1].

As we’ve previously said, the interesting behavior is not only about the transactions themselves, but rather about their use as links in a chain. For this reason, we define a “pseudo-spam chain” (PS-chain) as follows.

Definition 3.2. A *pseudo-spam chain* (PS-chain) is a sequence of PS-transactions in which the unique input address of the i -th transaction is the change address of the $(i - 1)$ -th transaction and the amount in input of the i -th transaction is the value payed to the change address in the $(i - 1)$ -th transaction.

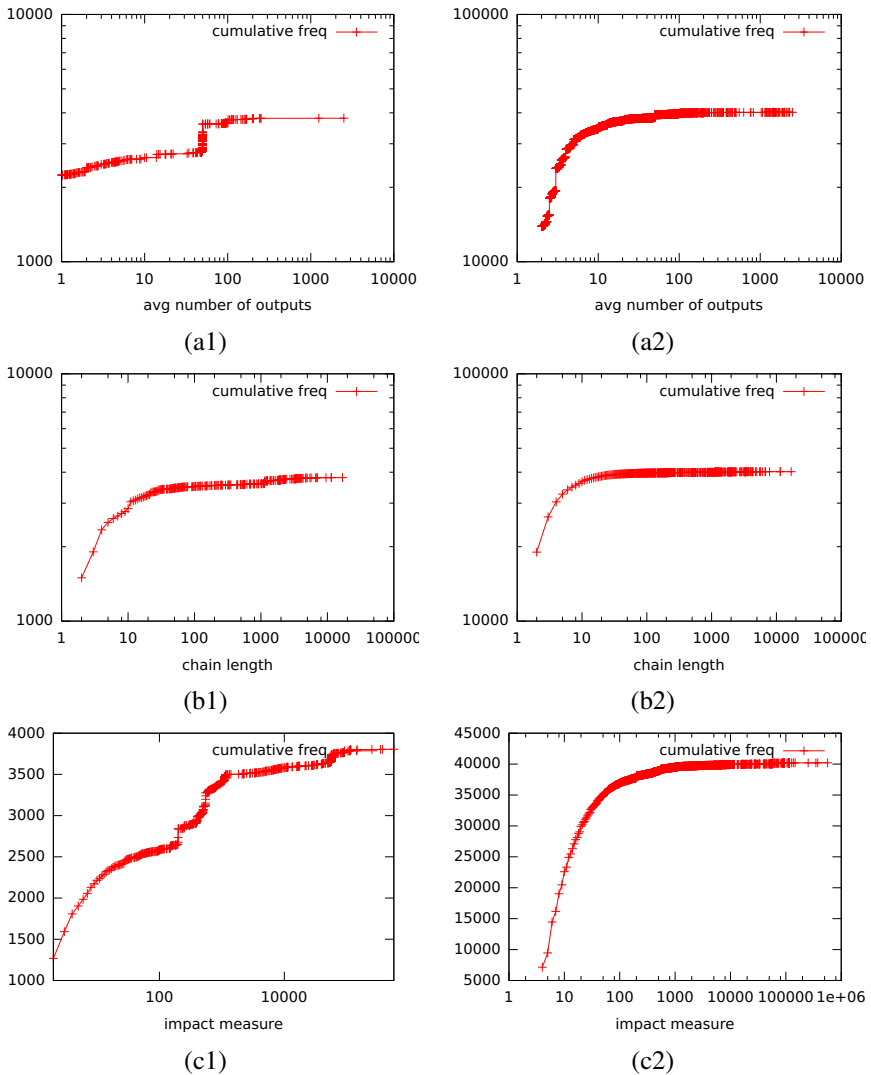


Fig. 2: PS-chains statistics (a1,b1,c1) and almost PS-chains statistics (a2,b2,c2)

Given a set T of PS-transactions, not always there exists just one pseudo-spam chain candidate including all the PS-transactions in T . We consider the smallest partition of T in PS-chain, i.e. whenever two PS-chains can be merged we merge them.

Considering as T the set of all the PS-transactions, we merged the PS-transactions in chains obtaining 24 381 PS-chains. To prune the PS-chains multiset from false positives we eliminated from the multiset all the singletons, hence discarding all the pseudo-spam transaction candidates that were not part of any chain. This left us with

3 805 PS-chains. In the following we report some basic statistics of the PS-chains we have found.

Average Number of Outputs. We have observed the cumulative distribution of the average number of outputs (excluding the change address linking to the next link in the chain) in each chain, which is shown in Figure 2(a1). We can notice how a large number of chains (i.e. 2 240) has exactly one single output address (excluding the change address). If we consider the cumulative distribution ignoring this special case, hence ignoring single output chains, we can notice a steep increase around 50: this value seems to be the preferred average number of outputs of the chains. The transactions we have manually examined had 100 outputs excluding the change address. We have seen that a good percentage of the PS-chains found share this behavior (approximately 7% if we don't count the single output ones).

Chain Lengths. If we consider the distribution of the lengths of the PS-chains found, shown in Figure 2(b1), we can notice very high initial values as well. More precisely, the chains of length two are 39.3%, while the chains of length at most three are already more than 50%.

Chain Impact. The small average number of output and the short length of many chains show that we found a lot of chains with a very low overall number of outputs. We define the *impact* of a chain as its average number of outputs times its length, or in other words the total number of outputs (excluding change addresses used as intermediary chain links) of all the transactions included in the chain. The cumulative distribution of this new measure is shown in Figure 2(c1). The higher this value is, the more “disruptive” the chain can be considered for the graph. From the plot we can immediately observe as 33.3% of all the chains have the minimum value, it means that one third of all the chains has length two and only one output is not a change output in each of its two transactions. We think that there is an high chance that these chains are normal and not artificially intended. Hence, for small values of this new measure we cannot label those chains as artificial since they may as well result from a lot of “normal” use cases. Even if those chains are not naturally occurring but deliberately created, their impact on the network is limited and not statistically relevant (since they represent 0.026% of all the multi-input multi-output transactions). We can chose a threshold for the chain impact measure, below which the chain are to be considered indistinguishable from legitimate transactions chains and we can prune the PS-chains set accordingly.

4 From the case study to generic chains

In the previous section we have defined what is a PS-transaction and a PS-chain starting from manual observations. The definition of a PS-transaction was given keeping into account our practical observation that such transactions payed an amount equal to 0.00001 BTC to each output, but we can easily observe that the amount payed as output is not the distinctive feature of the chains we're trying to model, their structure rather is. Taking into account this observation, we consider different

transactions sharing a similar structure to the PS-transactions but having arbitrary amount spent by the common outputs.

Definition 4.1. Given a transaction t we say that $t = (In, Out, InAmount, Fees)$ is an *almost PS-transaction* if it satisfies the following properties:

- $|In| = 1$;
- $|Out| \geq 3$;
- $|\{(o, b) \in Out : b \neq a\}| \leq 1$, for some $a \in \mathbb{R}$.

It is worth observing that not all the PS-transactions are almost PS-transactions. Indeed, we point out that we need to consider transactions of at least three outputs to be able to distinguish between the regular outputs and an eventual change address. Moreover, we also further restrict ourselves to only consider almost PS-transactions with common output value smaller than 1 BTC because high value transactions are more likely to be considered not spam.

We then define an almost PS-chain exactly as in Definition 3.2 but using almost PS-transactions instead. Applying our classification to the blockchain we found 1 050 783 almost PS-transactions that could be joined in 149 328 almost PS-chains. Among these, 40 208 almost PS-chains were not singletons.

If we perform the same analyses on some basic statistics of the almost PS-chains found as we did before for the PS-chains in Section 3, we obtain similar results. The plot of the cumulative distribution of chain lengths shown in Figure 2(a2) and number of outputs (excluding change addresses) shown in Figure 2(b2) show the same behavior as in Section 3, with 47.3% of the chains having length two and 34.6% of the chains having the minimum number of outputs. This suggests that our case study was a good approximation of the general phenomenon. If we evaluate the chain impact measure cumulative distribution as before we obtain a similar but smoother plot, shown in Figure 2(c2). For the almost pseudo-spam case we can also consider a new parameter that is the common outputs amount value of transactions and chains. The common output amount value cumulative distribution for almost PS-transactions found is depicted in Figure 3(a). We can immediately observe how the common output amount value used in our case study (0.00001 BTC) in Section 3 is the most frequent value for almost PS-transactions, covering 43.8% of all such transactions. We also note that all of the highest frequency common values are all “clean” values (for example 1, 600, 1000, 1250, 2750, 3000, 3500). This is compatible with human designed transactions rather than random purchase transactions, since prices are usually expressed in traditional fiat currencies such as USD or EUR, and their change in BTC is rarely a “clean” number. In Figure 3(b), we show the common output amount values cumulative distribution of the almost PS-chains found. In this graph the highest frequency values are clean numbers (1000, 7800, 10000, 100000, 200000, 500000, 1000000) as in the previous graph but the value 0.00001 has a smaller importance. This happens because a lot of the transactions with this common output value were joined in single long transactions.

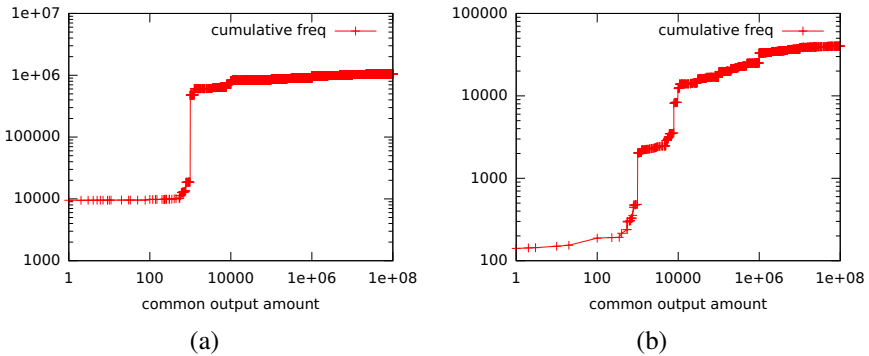


Fig. 3: Common output amount (expressed in 10^{-8} BTC) cumulative distributions for almost PS-transactions (a) and almost PS-chains (b)

5 Verifying Conjecture 3.1

In the following, we aim to prove Conjecture 3.1 by proving that the four outliers observed at the beginning of Section 3 are caused by few PS-chains targeting a small set of addresses artificially increasing their corresponding cluster's indegree. It is worth observing that not all of the output addresses of PS-chains are among those four outliers. Those other addresses do not stand out because they are part of already popular clusters, and so their indegree is marginally affected by those transactions while the pseudo-spam effect is more visible in other unpopular addresses. We also observe that we shall not expect all of the clusters with an indegree value of 708, 709, 771 or 772 to be artificially inflated. In fact, it is natural to expect the existence of a number of clusters, e.g. about 10, with these indegrees.

We start by checking if the clusters marked as outliers have at least one address that appears as output in a PS-chain. We find out that 1 630 over 1 647 clusters satisfy this. It means that only 17 clusters are not affected by the PS-chains. These findings are consistent with what we expected. More precisely, if we restrict just to cluster not involved in PS-chains we obtain an *outlier-free* indegree distribution. This alone is of course not enough to prove our supposition yet. We have only observed that all the outliers take part in a PS-chain but we still have to prove that the PS-chains are the sole cause of those outliers. To do so we firstly introduce the PS-set notion.

Definition 5.1. Given an almost PS-chains set C and a threshold $r \in \mathbb{R}$ we define as *pseudo-spam set* (PS-set) the set of transactions t_i such that there exists j with $t_i \in c_j$, $c_j \in C$, $|c_j| > 1$, and $\text{impact}(c_j) \geq r$, where $\text{impact}(c_j)$ is defined as the sum of the number of common outputs of each transaction $t_u \in c_j$.

In other words, given a threshold, a PS-set derived from an almost PS-chains set is the set of all the almost PS-transactions belonging to a chain in the candidate set that is not a singleton and has a chain impact measure greater than the threshold.

Now that we have a general definition for the artificial behavior we are trying to isolate we can finally verify whether Conjecture 3.1 is true or not. To check if a

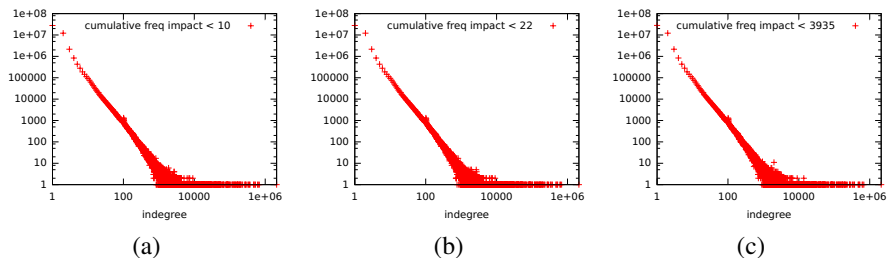


Fig. 4: Comparing the indegree distribution of the users graph pruned of the transactions belonging to the PS-set for threshold values of 10 (a), 22 (b) and 3935 (c).

pseudo-spam set alone is causing the indegree distribution outliers we re-compute the indegree distribution of the blockchain, ignoring the transactions belonging to the PS-set derived from the almost PS-chains candidate set obtained in Section 4, for increasing values of the threshold.

To choose the threshold values we look at the plot of the chain impact, shown in Figure 2(c2), and we observe that more than 50% of the chains have an impact value smaller than 10, more than 75% have an impact value smaller than 22 and more than 99% have an impact value smaller than 3935. So we choose those three values (10, 22 and 3935) to obtain a PS-set, this results in the indegree distributions depicted in Figure 4. As we can see the outliers disappear for all the values of the threshold considered without macroscopically affecting otherwise the overall distribution (see Figure 1(a) for a comparison). Not only this proves Conjecture 3.1 but it also means that the outlier generating chains of our case study are among the chains with largest impact, and so among the longest and with most outputs chains. This explains why those chains are the one that so macroscopically affect the indegree distribution of the entire network, enough to cause outliers in said distribution. Note that even if only the highest impact chains macroscopically affect the indegree distribution all the PS-transactions in the PS-set influence it. So also including lower impact chains helps cleaning the indegree distribution from artificial skewed values. Of course the lower the impact value used as threshold the more probable is the presence of false positives in the set, so a trade-of between the two has to be found.

6 Conclusions

This paper investigates the possible reasons of the presence of outliers in the indegree distribution of the bitcoin users graph. We have conducted an extensive set of analyses which have shown that the outliers are generated by artificial chains of transactions. We plan to extend our work to analyse other characteristics of the users graph. For instance, we are investigating whether the high diameter of this graph is due to other kinds of artificial transactions and we also plan to give insights into the nature of these transactions. More precisely we plan to further study the possible semantic of

PS-chains and to expand the analysis to include new types of artificial transaction patterns and their effect on the bitcoin users graph.

References

- [1] Bitcoin wiki, retrieved 18 sept 2016. URL https://en.bitcoin.it/wiki/July_2015_flood_attack
- [2] bitcointalk, retrieved 18 sept 2016. URL <https://bitcointalk.org/index.php?topic=458934>
- [3] reddit, retrieved 18 sept 2016. URL https://www.reddit.com/r/Bitcoin/comments/1xenyd/just_received_weird_tiny_payments_1sochi_1enjoy/
- [4] Androulaki, E., Karame, G., Roeschlin, M., Scherer, T., Capkun, S.: Evaluating user privacy in bitcoin. In: Financial Cryptography and Data Security - 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers, pp. 34–51 (2013)
- [5] Boldi, P., Rosa, M., Vigna, S.: Hyperanf: Approximating the neighbourhood function of very large graphs on a budget. In: Proceedings of the 20th international conference on World wide web, pp. 625–634. ACM (2011)
- [6] Borassi, M., Crescenzi, P., Habib, M., Kusters, W.A., Marino, A., Takes, F.W.: On the solvability of the six degrees of kevin bacon game - A faster graph diameter and radius computation method. In: Fun with Algorithms - 7th International Conference, FUN 2014, Lipari Island, Sicily, Italy, July 1-3, 2014. Proceedings, pp. 52–63 (2014)
- [7] Di Francesco Maesa, D., Marino, A., Ricci, L.: Uncovering the bitcoin blockchain: an analysis of the full users graph. In: IEEE DSAA 2016 Proceeding of 3rd IEEE International Conference on Data Science and Advanced Analytics, Montreal, Canada, October 17-19 (2016) (2016)
- [8] Fergal, R., Harrigan, M.: An analysis of anonymity in the bitcoin system. In: Proceeding of 2011 PASSAT/SocialCom 2011, pp. 1318–1326. IEEE (2011)
- [9] Kondor, D., Pósfai, M., Csabai, I., Vattay, G.: Do the rich get richer? an empirical analysis of the bitcoin transaction network. *PloS one* **9**(2), e86,197 (2014)
- [10] Lischke, M., Fabian, B.: Analyzing the bitcoin network: The first four years. *Future Internet* **8**(1) (2016)
- [11] Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M., Savage, S.: A fistful of bitcoins: characterizing payments among men with no names. In: Proceedings of the 2013 Internet Measurement Conference, IMC 2013, Barcelona, Spain, October 23-25, 2013, pp. 127–140 (2013)
- [12] Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2008)
- [13] Ron, D., Shamir, A.: Quantitative analysis of the full bitcoin transaction graph. In: Financial Cryptography and Data Security - 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers, pp. 6–24 (2013)

Networks with Hierarchical Structure: Applications to the Patent Domain

Nikolai Nefedov

Abstract In this paper we introduced a graph-based metric to measure a similarity between weighted sets of classifications codes defined as nodes on hierarchical taxonomy trees. We applied this metric to build relationship networks among companies and to find company peers (communities) in IPR (intellectual-property rights) domain based on patent portfolios.

To characterize evolution of patent portfolios for companies we used weighted sets of international patent classification codes (IPC), where each IPC weight corresponds to a number of IPC codes in a company patent portfolio aggregated to a given hierarchy level over a given period of time.

We used the suggested graph-based similarity at different hierarchical IPC levels to build corresponding networks and detected communities over different time periods. To track communities evolution in time we developed a cluster-matching algorithm to align community labels over time. Then we study evolution of communities in time to identify changes in a company strategy and its peers at the given time.

The suggested methodology may be applied to other domains that include hierarchical classification sets such as trademarks, legal documents, scientific papers, lawsuits etc.

1 Introduction

1.1 Patent networks

Patent network analysis is widely used to identify technology trends and formulate a technology strategy of a company, e.g., [1]. Typically patent networks are built using relationships among individual patents based on patent citations [2] or text analysis of patent abstracts, specifications, claims etc [3]. Recently patent text analytics is

Nikolai Nefedov (e-mail: nikolai.nefedov@thomsonreuters.com)
Thomson Reuters Labs, Switzerland and Swiss Federal Institute of Technology, Zurich (ETHZ), Switzerland

extended by using weighted keyword-based patent networks [4]. These methods usually are based on pairwise comparison of single patents complimented with total amount of patents in different technology sectors that allows to identify technology trends. On the other hand, in order to formulate a company strategy it is also important to know about activities of competing companies (peers) in relevant technology domains.

Finding company peers implies a comparison of profiles of companies and several attempts have been made to create company profiles or "fingerprints" reflective of assets and endeavors of the company. This may be done in several dimensions, e.g., fingerprint dimensions may include patent portfolio, trademarks, as well as products, fundamentals, geography, market associations, etc. At such fingerprints different taxonomy schemes (e.g., sets of classification codes) are widely used to describe dimensions. In this paper we address only patent portfolio domain.

Comparison of companies in IPR domain requires comparison of patent portfolios which include different amount of patents (patent weights) in different (and not necessarily overlapping) IPC categories. Besides, companies may have large patent portfolio volumes that makes difficult to differentiate and identify changes of topics using patents citations or patents text analytics. In this paper we used hierarchical International Patent Codes (IPC)[5] that are assigned by patent examiners and cover content of patents in more than 100 countries. Currently hierarchical IPC codes contain 8 sections (one letter), 130 classes (2-digit number), 639 subclasses (one letter), 7434 groups with 65152 subgroups (one-to-three digit number). In the following we refer these hierarchy levels h_k by number of symbols they contain, i.e., IPC1, IPC3, IPC4, IPC7. To compare patent portfolios we need to define a similarity between weighted sets of hierarchical objects.

1.2 Similarity measures

Similarity is widely used concept and many similarity measures have been suggested [6]. For example, a semantic measure in an IS-A taxonomy based on a shared information content of the shortest common distance between two words/concepts in a lexical taxonomy is proposed in [7, 8]. As its generalization, an universal definition of similarity from information theory point of view was developed in [9]. However, these concepts mainly address a similarity between single objects, while to compare patent portfolios we need to define similarity between sets of weighted hierarchical elements. On the other hand, methods to calculate similarity between sets of objects typically do not take hierarchy into account (e.g., cosine similarity).

In this paper we propose a similarity measure to compare weighted sets of hierarchical objects and applied it for patent portfolios comparison. The proposed similarity measure allowed us to present relations among objects, e.g. companies, as a *connected* graph; it is hardly possible with other types of similarity such as cosine similarity. Then we applied network analysis to find peers and analyze peers evolution in time. Also, the proposed method allows us to map activities of companies on a connected technology map to provide a view on a broader technology evolution.

The paper is organized as follows: Section 2 outlines a graph-based metric to compare weighted hierarchical sets. In Section 3 we built patent portfolio evolution for a number of companies at different hierarchical IPC levels. Next we used the suggested metric to calculate pairwise similarities between companies in IPR domain at different hierarchical levels followed by construction of corresponding networks and their evolution in time. To find peers (communities) we applied community detection methods [10, 11, 12] at different IPC hierarchical levels h_k for different years (2008-2014). To track communities evolution in time we developed a cluster-matching algorithm to align community labels over time based on [13]. Finally, we analyzed evolution of communities in time to identify changes in a company strategy and its peers at a given time.

2 Comparison of weighted hierarchical sets

2.1 Preliminaries

Let's consider a set C of objects c_i , where $|C| = N_c$ is a total number of objects. Relations between objects $\{c_i, c_j\}$ may be presented as a weighted undirected graph $G(C, E, \mathbf{S})$, where $E = \{e_{ij}\}$ is a set of edges $e_{ij} \in \{0, 1\}$ and \mathbf{S} is a similarity matrix, $s_{i,j} = s(c_i, c_j) \in \mathbf{S}$, $i, j = 1, \dots, N_c$, is similarity between c_i and c_j . Hierarchical attributes for a given object c_i may be presented as a tree $T_i(\mathbf{a}(h_k))$, where c_i is the the root and attributes $\mathbf{a}(h_k)$ are nodes of c_i on the tree at a hierarchical level h_k . As an example, let's consider objects c_1 and c_2 with attributes taken from a set $\mathbf{a} = \{A, B, C, D, E, F, G, H\}$ corresponding to IPC1 as shown at Fig. 1. Similarity between objects c_i and c_j (shown by dashed lines) usually is defined as a function of intersection of corresponding subsets $\mathbf{a}(c_i)$ and $\mathbf{a}(c_j)$, e.g., $s(c_i, c_j) = f|\cap(\mathbf{a}(c_i), \mathbf{a}(c_j))|$ (cf. Fig. 2).

In the following we will call relations graph $G(C, E, \mathbf{S})$ as a network to avoid confusion with graphs presenting taxonomy trees T_i .

2.2 Weighted taxonomy trees

Figure 2 illustrates the suggested approach to define relationships between objects c_1 and c_2 with weighted hierarchical attributes at levels IPC1, IPC3 and IPC4. In case of patent portfolios, weights $w_n(h_k)$ may present a number of IPC codes aggregated to level h_k within considered IPC class (B02F, B02,B at Fig.2). Let's assume that objects c_1 and c_2 have, among others, patents in IPC code B02F, Fig. 2. Then this IPC category contributes to similarity $s(c_1, c_2)$ at three hierarchical levels $\{B, B02, B02F\}$ (see dashed lines between c_1 and c_2) such that the deeper we go down on the tree, the higher similarity is: $s(c_1, c_2, h_1) < s(c_1, c_2, h_2) < s(c_1, c_2, h_3)$. For example, if we compare IPC classes B02G and B02F, then for these codes only 2 layers $\{B02, B\}$ contribute to similarity; note no similarity between B0G2 and F04.

Generalization to weighed hierarchical sets and its applications is briefly outlined below. In particular, a patent portfolio for a company c_j may be presented as a set of tuples $P_j(h_k) = \{a_i(h_k), w(a_i(h_k))\}$, where $a_i(h_k) = IPC_i(h_k)$ is the i -th IPC code in patent portfolio at the k -th hierarchy level, $w(a_i(h_k))$ is its weight, $i \in N_j(h_k)$ is a number of different IPCs in $P_j(h_k)$. In our case $w(a_i(h_k))$ is a number of IPCs aggregated from all patents containing $IPC_i(h_k)$ code. Note that since there may be multiple IPCs characterizing a single patent, this definition applies both to patent portfolios and to single patents. In the following we call tuples $P_j(h_k)$ as aggregated IPCs at the level h_k . For example, patent portfolios aggregated to $h_k = 3$ level and sorted by weight for companies $c_1 =$ 'Samsung Electronics' and $c_2 =$ 'Panasonic' are presented as $P_1(3) = \{\{G06F, 10251\}, \{H04N, 7800\}, \{H01L, 6634\}, \dots\}$. and $P_2(3) = \{\{H04N, 5920\}, \{G06F, 4989\}, \{H01M, 2616\}, \dots\}$, respectively.

2.3 Similarity between weighted hierarchical sets

Typically methods to calculate similarity (e.g., cosine similarity) do not take hierarchy into account. For example, cosine similarity between patents having rather similar IPC codes A01B11 and A01B12 is zero. Similar to patent portfolios comparison, the problem exists in patent to patent comparison since even a single parent may be categorized by a set of IPC codes. Furthermore, it is not clear how to take into account weights at different hierarchical levels and define a normalization to compare *weighted sets* of hierarchical classification codes, such as patent portfolios with multiple IPCs. In this section we briefly outline the proposed method to compare weighted sets of hierarchical objects where sets have the same cardinality. More detailed generic description of the proposed method is rather involved and to appear elsewhere.

Let's define $p(a_l, c_i) = [a_l, \dots, c_i] = p(a_l^i)$ as a sequence of nodes on T_i forming the shortest path from node a_l to root c_i . Then we may define a similarity s between nodes a_l and a_m as a number of common nodes between paths $p(a_l^i)$ and $p(a_m^j)$:

$$s(a_l, a_m) = s(p(a_l^i), p(a_m^j)) = \left| \bigcap (p(a_l^i), p(a_m^j)) \right|. \quad (1)$$

Clearly, $s(a_l, a_l) = |p(a_l)|$ corresponds to a number of hierarchical levels on the path from a_l to the root on T_i . Similarly, $s(a_l, a_m)$ may be seen as a number d of shared hierarchical levels or a distance $d(a_l, a_m)$ on T . In this settings s is a linear function of d . On the other hand, for irregular trees such as IPCs taxonomy, contributions to similarity may not necessary depend linearly on h_k . To take this property into account we included function $f(h_k)$ into the normalization below. Recall that the longer a classification code, the more information it provides, i.e., $s(h_k)$ is a monotonically increasing function of h_k .

Let \mathbf{a} and \mathbf{b} be portfolios for companies c_1 and c_2 . Then a normalized similarity s_n between two codes from \mathbf{a} and \mathbf{b} on the same taxonomy tree may be written as

$$s_n(a_l(h_k), b_m(h_k)) = \frac{s(a_l(h_k), b_m(h_k))}{f(h_k)}. \quad (2)$$

It may be shown that a normalized similarity between unweighted hierarchical sets **a** and **b** at level h_k may be presented as below

$$s_n(\mathbf{a}, \mathbf{b}, h_k) = \frac{1}{C_{max}} \sum_l^N \sum_m^N s_n(a_l(h_k), b_m(h_k), f(h_k)), \quad (3)$$

where

$$C_{max} = 1 + (N - 1)f(h_{max} - 1) / f(h_{max}). \quad (4)$$

A normalized similarity between weighted hierarchical sets **a** and **b** (patent portfolios) aggregated to h_k level may be written as

$$s_n^{(w)}(\mathbf{a}, \mathbf{b}, h_k) = \frac{1}{C_{max}^{(w)}(f, N, h_{max})} \sum_l^N \sum_m^N \Phi(w_l^{(a)}, w_m^{(b)}, W^{(a)}, W^{(b)}) s_n(a_l, b_m, f(h_k)). \quad (5)$$

Note that there may be different ways to define function $\Phi()$. For example, by applying the same methodology as in (1) for weights we may derive a weighting symmetric function as below

$$\Phi(w_l^{(a)}, w_m^{(b)}, W^{(a)}, W^{(b)}) = \min\left(\frac{w_l^{(a)}}{W^{(a)}}, \frac{w_m^{(b)}}{W^{(b)}}\right) \quad (6)$$

$$W^{(i)}(h_k) = \sum_m w_m^{(i)}(h_k), \quad i = a, b \quad (7)$$

The max similarity in (Eq.5) is reached when all IPC codes in both portfolios are located in the same IPC class at the lowest hierarchy level.

This methodology may be extended to comparison of two ontologies with a difference that instead of a single underlying tree as in the case above, there may be several (or a forest of) underlying trees. It implies that mapping of ontology objects and similarity calculations should be aggregated over relevant subsets of underlying trees.

3 Patent portfolios comparison

3.1 Evolution of patent portfolios

Companies change direction and enter new areas of technology and may cease operating in long-involved areas of technology. In this section we analyzed evolution of company patent portfolios at different hierarchical levels to detect changes in a company activities. As a data source we used Derwent Patents Database [14]

available via *Thomson Innovation*[15] and built patent portfolios for 10^5 companies covering totally about 3×10^6 patent families registered in the USA during period 2008-2014.

As an example, Fig. 3 shows IBM patent portfolio evolution at different hierarchical IPC levels over time. Here colors correspond to different IPC codes for patents within IBM patent portfolio, labels on side color-bars indicate patents mapping to the highest hierarchical level IPC1. The absence of patents in a particular IPC category is denoted by blue color (black color in paper version). For example, one can notice a blue color line during 2010-2014 at Fig. 3a ($y=19$ corresponds to IPC3 = G07) and Fig. 3b ($y=38,39$ correspond to IPC4 = G07C,G07F). It indicates that IBM stopped patent activity in measurement equipment for registering tokens. On the other hand, from 2010 there is growing activity in IPC3=B81 ($y=8$ at Fig. 3a) corresponding to nano-technology, in particular, in field of manufacturing of devices and systems on substrate IPC4=B81C (Fig. 3b).

Note that new trends may not easily be observed at a very coarse or a very granular hierarchy levels, so we used cross-level analysis to detect changes and then digging for more details.

3.2 Networks evolution

Networks are dynamic and changing over time with some companies becoming peers and other peer companies losing the association as a peer company due to a number of reasons. Over time companies enter the competitive landscape and fall out of the landscape. Dynamic network analysis and models to describe evolution of communities are under intensive studies, in particular, in social networks (SN) domain, e.g., [16, 17, 18]. In this paper we do not consider models for SN communities evolution, but primary addressing a discovery mode to look for disruptive changes that modifies competition profile in IPR domain.

In particular, given sets of classification codes (e.g., IPC-based patent portfolio) defined on the same classification tree we analyzed peers (communities) evolution using the following steps:

- (a) define graph-based similarity metric as a function of distance between nodes on the underlying classification tree;
- (b) calculate pair-wise similarity between nodes by mapping nodes (IPCs) from different portfolios to the underlying classification tree (see Eq.2);
- (c) calculate similarity metric between sets of weighted classification codes (e.g., general case Eq.5, examples Eq.6, Eq.7) and build network snapshots for different time periods;
- (d) apply community detection algorithms to network snapshots to find stable communities based on random walk [12] within each time snapshot;
- (e) build a reference network by aggregating all network snapshots over time and applied community detection algorithms to find communities within;

- (f) use aggregated community labels as a reference and matched community labels from different network snapshots to the reference community labels;
- (g) steps above allow us to analyze communities evolution over time, detect company peers at given time and predict new trends.

Fig. 4 shows a network example built using 10 IPC codes with largest weights in each patent portfolio for the top 300 companies with largest patent portfolio volumes. We found that the suggested method results in a connected network, but for visualization purposes Fig. 4 shows only 5% of largest similarity values. As one can see, even under this simplification, the suggested method results in several connected clusters which allows to find mapping to technology areas and its relations. Also it easy to detect companies which are active in several technological areas, such as 'Siemens', 'Samsung', 'Hitachi Chemical' and 'Funai Electric'.

Fig. 5 presents an example of evolution of peer communities in time before (on the left) and after (on the right) community labels matching for the top 100 companies with the highest patent portfolios volumes. The first column on the left on both figures shows references for communities matching. All nodes (company IDs) in time snapshots are grouped according to the reference layer grouping.

As one can see from Fig. 5b, the largest part of competitive landscape stays mainly stable (shown by yellow in online version), while some companies are moving or exploring other technology domains. At the same time one group of companies (green in online version) keeps investing in another technology domain (orange in online version) in 2009 and 2013, while staying in its main domain the other time.

4 Conclusion

In this paper we propose a similarity measure to compare weighted sets of hierarchical objects. As an example, we consider company patent portfolios characterized by hierarchical IPC codes. Using the suggested similarity measure we build network snapshots for different time periods and applied network analysis to find company peers in IPR domain. It allows us to study peers evolution at different hierarchical levels and find changes in competitive landscape. The suggested methodology may be applied to other domains that include hierarchical classifications.

Acknowledgements This work was supported by Thomson Reuters Global Resources. The author would like to thank anonymous reviewers for comments and pointing to missing references.

References

- [1] Valverde S. et al, Topology and Evolution of Technology Innovation Networks. *Phys. Rev. E* 76, 056118 (2007).
- [2] Verspagen B., Mapping technological trajectories as patent citation networks: A study on the history of fuel cell research. *Advances in Complex Systems*, 10(1), 93–115 (2007).
- [3] Yoon B. et al, A systematic approach for identifying technology opportunities: keyword-based morphology analysis. *Technol. Forecast.* 72, 145–160, Elsevier (2005).

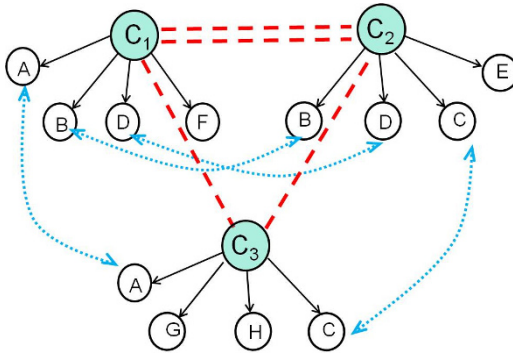


Fig. 1: Nodes with attributes.

- [4] Lee S. et al, An approach to discovering new technology opportunities: Keyword-based patent map approach. *Technovation* 29(6), 481–497, Elsevier (2009).
- [5] International Patent Classification, <http://www.wipo.int/classifications/ipc/en/>
- [6] Cha S–H., Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions. *J. Math. Models and Methods in Applied Sci.* 1(4), 300–307 (2007).
- [7] Resnik P., Using information content to evaluate semantic similarity in a taxonomy. *Proceedings IJCAI*, 448–453 (1995).
- [8] Resnik P., Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *J. Artif. Intel. Res.* 11, 95–130 (1999).
- [9] Lin D., An Information-Theoretic Definition of Similarity. *Proc. Int. Conf. on Machine Learning*, 296–304 (1998).
- [10] Newman MEJ, Fast algorithm for detecting community structure in networks. *Physical Review E* 69, 066133 (2004).
- [11] Blondel V. et al, Fast unfolding of communities in large networks. *J. Stat. Mech.: Theory and Experiment*, 1742–5468 (10), P10008+12 (2008).
- [12] Lambiotte R. et al, Laplacian Dynamics and Multiscale Modular Structure in Networks. *ArXiv:0812.1770v3*.
- [13] Nefedov N., Analysis of Communities Evolution in Dynamic Social Networks. *Studies in Computational Intelligence: Complex Networks IV*, 476, 39–46, Springer (2013).
- [14] DWPI: <http://ipscience.thomsonreuters.com/product/derwent-world-patents-index-dwpi>
- [15] <http://ipscience.thomsonreuters.com/product/thomson-innovation>
- [16] Palla G. et al, Quantifying social group evolution. *Nature* 446, April, 664–667 (2007).
- [17] Lin Y-R et al, Analyzing Communities and Their Evolutions in Dynamic Social Networks, *ACM Trans on Knowledge Discovery from Data.* 3(2), 8 (2009).
- [18] Brodka P. at al, GED: the Method for Group Evolution Discovery in Social Networks, *Soc. Netw. Anal. Min.* 3(1), 1–14 (2013).

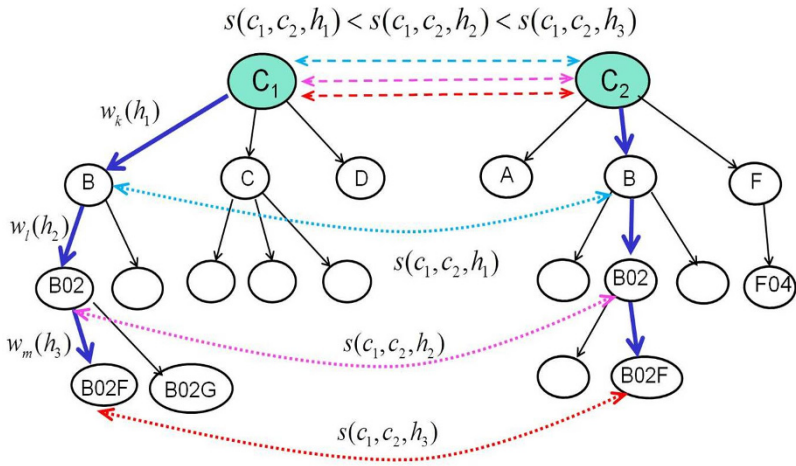


Fig. 2: IPCs as taxonomy trees.

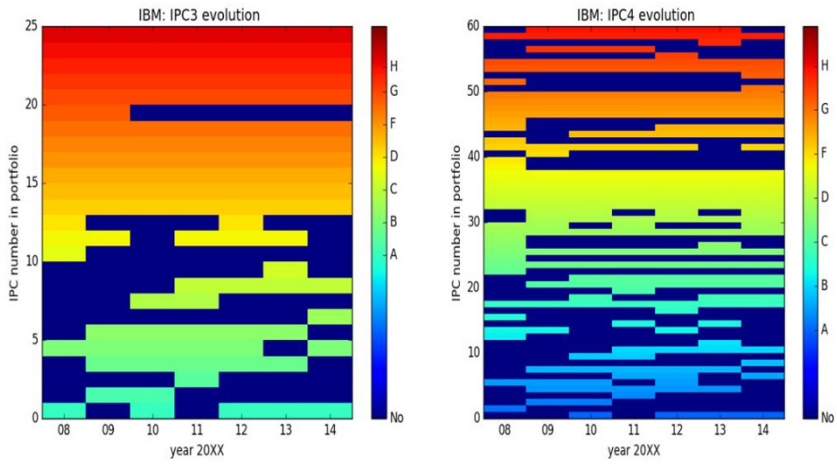


Fig. 3: Example of patent portfolios evolution in time at difference hierarchy levels. Company: IBM; hierarchical levels IPC3 (left, *a*) and IPC4 (right, *b*). Colored bars indicate mapping to the highest hierarchical level IPC1 (colored figures online).

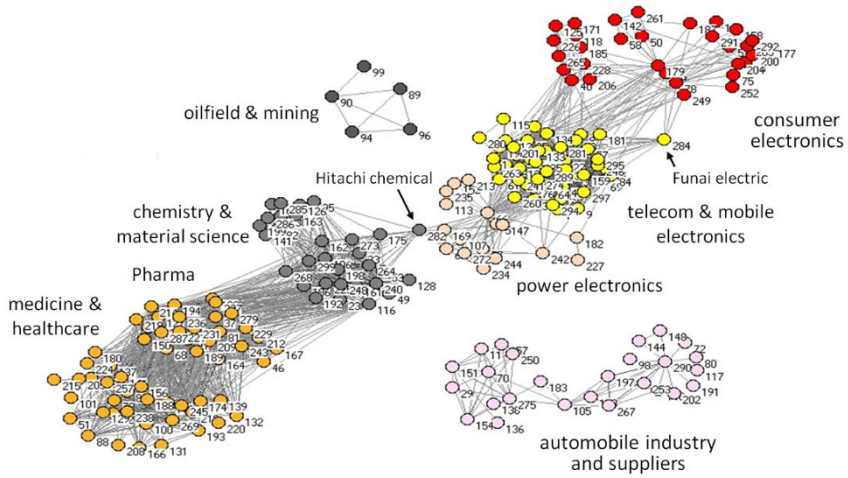


Fig. 4: Mapping patent portfolios of top 300 companies on technology categories: network with 5 % of strongest similarities to highlight technology categories; hierarchical level IPC4; 10 IPCs in each portfolio with the largest weight (colored figure online).

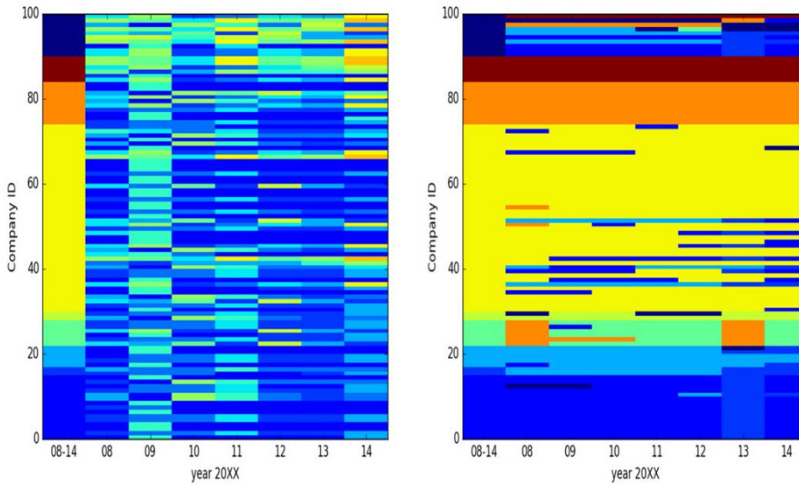


Fig. 5: Example of evolution of peer communities (shown by colors) in time before (left, *a*) and after (right, *b*) community labels matching for the top 100 companies with the highest patent portfolios volumes. The first column on the left on both figures is used as a reference for communities matching. This reference corresponds to communities detected in an aggregated network built over time period 2008-2014. All nodes (company IDs) in time snapshots are grouped according to the reference layer grouping.

Social Connection Dynamics in a Health Promotion Network

Eric Fernandes de Mello Araújo, Michel Klein and Aart van Halteren

Abstract The influence of social connections on human behaviour has been demonstrated in many occasions. This paper presents the analysis of the dynamic properties of longitudinal (335 days) community data ($n=3,375$ participants) from an online health promotion program. The community data is unique as it describes how the network has evolved since its inception and because the information exchanged through the network was predominantly about the achievements of participants in the program and therefore influencing behavior through social comparison. The analyses show that the largest component of the community network has characteristics of a small world network. The analyses also show that connections are formed according to a strong attachment preference according to the gender, and a weaker homophily for Body Mass Index. The presented analysis can serve as basis for creating novel interventions that influence physical activity behavior through social connections.

1 Introduction

Social Network Analysis (SNA) is a broad research area, with applications in many different disciplines, incorporating aspects of sociology, social psychology and anthropology [19]. SNA is useful for studying nodes' influences within a network, and how behaviours, opinions or sentiments are spread in social networks [3, 6]. The nodes with an important position can be used to find points of interventions to stop or to enhance the process under study [1, 2, 9, 11, 21].

However, many of the contributions in this field are based on static networks, without taking the time dimension into account. The dynamics of the network can reveal more about how the network evolves over time [5, 22].

Eric Fernandes de Mello Araújo (e-mail: e.araujo@vu.nl)✉ · Michel Klein (e-mail: michel.klein@vu.nl)✉
VU Universiteit, Amsterdam, The Netherlands

Aart van Halteren (e-mail: aart.van.halteren@philips.com)✉
Philips Research, Eindhoven, The Netherlands

In this paper, we investigate the dynamic properties of longitudinal (336 days) community data ($n=3,375$ participants) from an online health promotion program. This data set presents a network of people that share their physical activities and see others' activity levels. It is a data set specifically focusing on health promotion, in contrast with other research which is mostly using online social networks for general purposes, such as Facebook, Twitter, etc. [8, 15].

To build this data set, the participants wore an activity monitor device that tracks their physical activity level (PAL). They also had access to an online system where they could befriend other participants in order to share and see each others' PAL. The data sample used in this work was collected from 28/04/2010 until 30/03/2011. The analysis of the characteristics of this social network in a health promotion context provides a basis for answering the following questions:

1. How does the largest component of this specific social network develop over time?
2. Does this social network demonstrate the homophily phenomenon (concerning gender and BMI)?
3. Can we use the dynamic analysis of the network to determine influential nodes?

The paper is organized as follows. Section 2 discusses the dynamic aspects of social networks, and presents the concepts explored here. Section 3 explains the analysis performed, metrics used and the selection process. Section 4 shows the results of the analyses. Finally, Section 5 concludes the paper with a discussion of the consequences and the possible applications of the findings.

2 Dynamical Social Network Analysis

The dynamic aspects of social networks can be analyzed in two ways: (1) looking at the changes *inside* the network (changes in the nodes' attributes as opinions, beliefs, etc.), or (2) looking at the changes of the network itself (the topology of the network, the nodes' degrees, etc.). Dynamical networks are considered here as social networks where the topology changes over time due to new connections or new subjects inside the network.

Static measures of nodes' degrees, centrality, shortest paths, etc. of one fixed snapshot of the data are not sufficient to understand real networks that evolve over time. How new connections are made in or removed from the social network can to some extent be explained by these two phenomenons: homophily and preferential attachment ('more becomes more') [4, 14]. These concepts will be explored further in this work.

The dataset that we use is also used in [10]. In their work, the authors explore the internal states of the nodes and the correlations between the characteristics of the nodes for a shorter period (14 weeks). In [13], the same data set is the basis for a study on the differences between people inside and outside a community, showing how the community aspect plays a role in changing the physical activity level during an intervention. The current work is dedicated to the topological and structural aspects of the network and its connections over time.

3 Methods

This section explains the data collection and the data processing. The aim is to provide a clear understanding of how the data was collected, how the subset was selected and how the analysis was done.

3.1 Data Set and Data Selection

The data set is the result of an online physical activity promotion program, where the participants wore an activity monitor that tracks their physical activity level (PAL). The devices were synchronized with an online system, which also provided the possibility for them to join a community through connection requests. The participants could also participate in a health promotion program, and those who decided to do that were tagged in our data set with a ‘start plan date’. The data used in this work spans 336 days, from 28/04/2010 until 30/03/2011.

As the decision to join the community was optional for the participants, around 10% of them decided to join the social network to exchange their information about the PAL tracked by their devices. In total there are almost 5,000 nodes that opted to join the online community at some moment during the experiment.

Due to changes in the system, some cleaning was necessary to keep the data set reliable for the analyses performed. From the originally 5,000 nodes and around 28,000 edges, we filtered nodes and edges according to the following characteristics:

- a) Nodes without ‘start plan date’ were removed;
- b) Nodes were included according to the date of their started plan;
- c) Nodes that dropped out the experiment (tagged with a value for ‘dropout date’) were taken off at the day when they quit the network and the program;
- d) Nodes without a value for BMI (Body Mass Index), gender and nodes in which all information was missing were taken out;
- e) Edges without ‘start date’ value were removed;
- f) Edges connected to excluded nodes were removed.

From a total of 28,418 edges, 3,802 edges didn’t have information about the date of connection, because some requests for connections in the network were not approved from the receiving peer. As these edges are represented in two directions, 1,901 unique edges were discarded. From the 24,616 edges left, 12,047 are duplicated edges, i.e., node A connects to B, but the edge (B,A) already exists. As all connections are bidirectional, this is redundant data. So we have, in the end, a total of 12,569 edges representing connections that were formed during the experiment.

The data set originally contained 4,989 nodes. Of those, 1,614 nodes were not eligible because they do not have values for all the attributes needed for the analysis (i.e., gender, BMI and start plan date). The selected data set has 3,375 nodes left.

The nodes are only included in the network in the period between the start plan and the drop out date (for those that dropped out). After the node leaves the network, all its connections are deleted also. The impacts of the cleaning process are irrelevant, because the nodes and edges removed didn’t participated in the program as demanded.

3.2 Social Network Analysis

The network measures that are calculated are [19]: (1) degree distribution; (2) average degree; (3) closeness centrality; (4) eigenvector centrality; (5) betweenness centrality; and (6) average shortest path . These aspects were analyzed for each day of the experiment.

Formula 1 shows the calculation for the **combined centrality**, a combination of the betweenness and closeness values:

$$Comb_C(i) = \frac{C_C(i) + C_B(i)}{2} \quad (1)$$

$C_C(i)$ and $C_B(i)$ are the closeness and betweenness centralities, respectively. This formula doesn't consider the balance between the two centrality measurements, and might be improved for future analysis. For our analysis it is correct to say that the Closeness centrality will influence more than the betweenness for having higher values in general.

Homophily is the tendency of nodes to create strong connections with others that are alike, have the same opinions, or share similar characteristics [14]. The homophily principle can be studied in two ways: the *social* homophily and the *value* homophily [12, 20]. In this work, the *social* aspects (gender and BMI) are studied in depth, while the *value* aspects are left out of the analysis.

The homophily according to gender was calculated using the gender of the nodes' edges. These edges were categorized as follows:

Edge MM (EMM): a connection between two male nodes;

Edge MF (EMF): a connection between a male node and a female node;

Edge FF (EFF): a connection between two female nodes.

As the three categories are disjoint, the total number of edges equals to $EMM + EMF + EFF$. The homophily for female gender and male gender are given by equations 2 and 3, respectively.

$$Homophily_F = \frac{EFF}{EFF + EMF} \quad (2)$$

$$Homophily_M = \frac{EMM}{EMM + EMF} \quad (3)$$

To calculate homophily for the BMI, we considered nodes with BMI in the same range as equals. Two different thresholds were used: 5.0 and 6.5, which are the respective ranges for the group of Normal and Overweight BMI in the categorization according to [18].

The ratio between the nodes' edges with a small difference in BMI and the total number of edges yields the percentage that follows the homophily principle for the BMI. The equations follow the same principles of equations 2 and 3.

The **ego-network density** for the nodes is used to find important nodes. The density is calculated in two steps. First, the ego-network of all the nodes (including the observed node) is created using 1-step neighborhood. After this step, the density

of the ego-network was calculated as: Ego-density = $\frac{|E|}{n(n-1)}$, where $|E|$ is the number of edges in this subgraph, and n is the number nodes.

4 Results

This section presents the results obtained from the social network analysis. The section is organized according to the questions from Section 1:

1. How does the largest component of this social network develop over time?
2. Does this social network demonstrate the homophily phenomenon (for gender and BMI) ?
3. Can we use the dynamic analysis of the network to determine influential nodes?

4.1 Nodes, edges and degree distribution

On day 98 of the experiment the number of nodes in the graph is stabilized at 2,996. The number of nodes in the largest component increases until the end of the experiment, due to new connections established among the nodes.

For the edges there is also a point of stabilization in the new connections around day 100. From that day onward there is a very small increase in the number of connections (around 8.2%). Most of the edges are in the largest component, as it is expected in a network that follows the Small World Network model.

The graph follows a Power-law distribution for the degrees of the nodes for all time steps. Figure 1 shows the degree distribution for the days 1, 100 and 336 in a log-log scale (for illustration¹). The lower graphics show the coefficients for the linear regression of the correlation between the degree of the nodes and the number of nodes with certain degree.

As shown in the lower graphic, the p value is always significant for our data set, and the R-squared is close to 1, showing that the model explains very well the data, mainly after day 100.

The ‘more becomes more’ principle is the assumption that nodes with higher degree have a higher chance of receiving more connections over time [16]. Figure 2 shows how the degrees of the nodes with the fewest connections (the ‘poorest’, right) and nodes with the most connections (‘richest’, left) evolve over time. More investigation is needed to claim that the preferential attachment is observed here, but the information about the rich and poor nodes suggests that it could be present in our data set.

4.2 Largest component and other components

The ‘largest component’ is the biggest connected component among all components of any graph. Figure 3 shows the percentage of the nodes of the graph that are part of the largest component for all time steps in two different scenarios. In the first scenario, all nodes are included in the graph. As can be observed, the average number of nodes in the largest component is 65% after day 296 for the entire graph. The increase in the percentage follows the inclusion of new edges after time 100 (when the number of nodes is stable).

As there are many nodes with degree 0 (isolated nodes), for the second scenario, the nodes with degree 0 were excluded from the graph. In this scenario the percentage of nodes in the largest component goes up to 80%.

¹ The other days and other animations can be seen at <http://www.cs.vu.nl/~efo600/cn2016/>

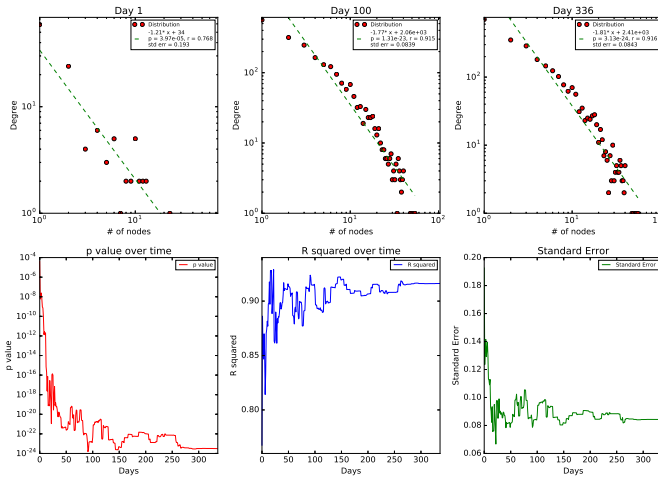


Fig. 1: Degree distribution in days 1, 100 and 336 (top) and p value for slope, R squared and standard error (bottom)

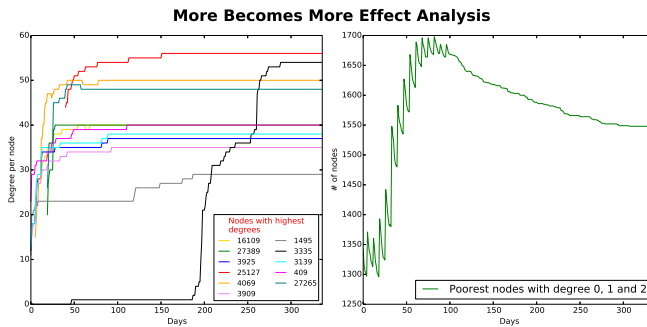


Fig. 2: Degree of the richest nodes (left) and number of poor nodes, with $degree \leq 2$ (right)

Figure 4 shows the evolution of the connected components over time. The upper graphic shows the number of components over time. As edges are inserted, many components are joined, explaining the decrease from around 1,200 connected components to almost 600 in the end. The red line shows the number of components bigger than 1, i.e., non isolated nodes. This number goes from 39 on day 1 up to 164 in the last day of the experiment. The number of isolated nodes goes from 1,193 on day 1 down to 492, what explains the high number of components, even after the largest component gathered more than 60% of the nodes of the network.

The correlation between the size of the components and the number of components with a specific size (frequency of occurrence) is shown in the middle part of Figure 4 in three graphics, for days 1, 165 and 335. The correlation is significant for all time steps. The three lower graphics show the p value, R squared and standard error for the regression done in all the time steps of the data set. It can be seen that the fit parameter goes from approximately 65% to less than 40% in the end of

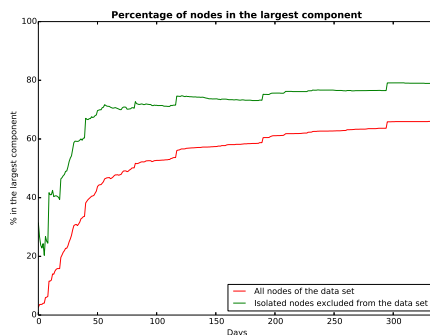


Fig. 3: Percentage of the nodes in the largest component. All nodes (lower red line) and nodes with degree larger than 1 (higher green line)

the experiment. This can be explained by the changes in the largest component, and the joining of previously separated components.

4.3 Centrality measurements

As the largest component has most of the nodes and edges, it is also interesting to explore the centrality measurements for this component. The following metrics were analyzed: (1) betweenness centrality, (2) closeness centrality, (3) eigenvector centrality, (4) average shortest path.

The **betweenness centrality** indicates how important a node is for the transfer of information or any kind of spreadable element inside a network. Nodes with higher betweenness have more shortest paths passing through themselves, and therefore can enhance their role in the network. The **closeness centrality** is the proximity of a node to the rest of the network, and it is calculated by the inverse of the sum of the shortest distances between each node and all other nodes in the network. The **eigenvector centrality** is calculated based on the centrality of its neighbors.

The average centrality for all the nodes (betweenness, closeness and eigenvector) is shown in Figure 5. The first three graphics on the left show all time steps, while the first three graphics on the right provide a zoomed-in version between day 50 and 336.

The lower graphic in Figure 5 shows the average shortest path. The average shortest path for our data set stabilizes around 6.5, a low value as suggested by the theory in [17].

The combined centrality is useful in finding important nodes that combine a good betweenness centrality and closeness centrality. Figure 6 shows the combined centrality for all the nodes with degree higher than 1.

It is possible to highlight the list of nodes with higher centrality (the most potentially influential nodes in the network). Figure 6 shows the most central nodes measures of betweenness, closeness and the combined centrality. As shown in Figure 6, nodes 68593 and 3335 are very important for this data set, as they present the highest values for these measurements.

4.4 Homophily

To investigate homophily according to gender and BMI, the edges were evaluated to determine whether the nodes they connect belong to the same category. The results for the gender analysis follow the equations 2 and 3. The data set has 51.4% of the nodes of gender male, and 48.6%

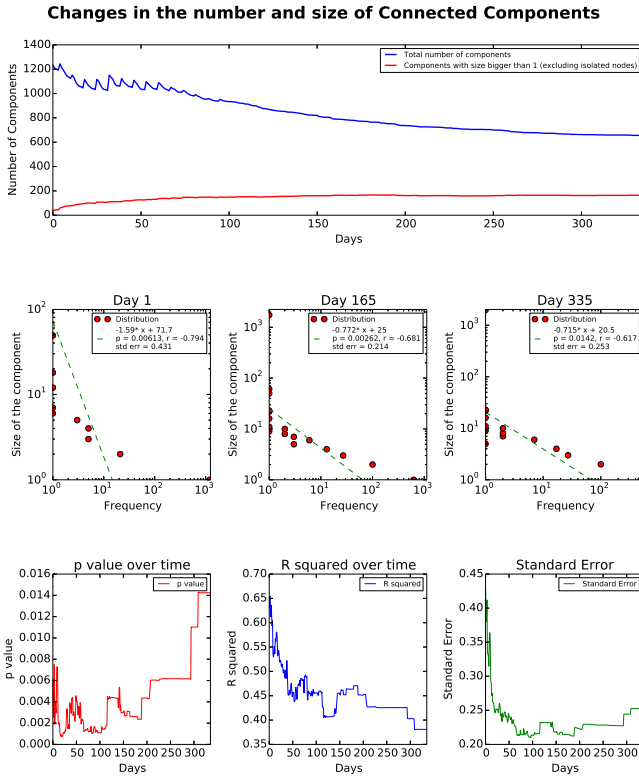


Fig. 4: Components analysis. Number of components in the graph over the time (upper), the correlation between the size of the component and the frequency of the size (days 1, 165 and 335) (middle) and the parameters from the linear regression for all time steps (lower)

female. Regarding the BMI of the population, 0.8% are underweight, 33.6% are normal, 34.8% are overweight and 30.8% are obese [18].

Figure 7 (left) shows the homophily according to the BMI of the nodes. Two ranges were tested for the nodes: 5.0 and 6.5. For the range of 5.0, the ratio of edges with nodes within the same range is around 50% after day 100, while for range 6.5 this value is increased to around 59%. For both ranges, more than half of the connections are within nodes with close BMI.

Figure 7 (right) shows the homophily according to gender. Three calculations were made: (a) edges connecting male-male nodes, (b) edges connecting female-female nodes and (c) edges connecting same gender nodes (male-male plus female-female edges). In this data set, the homophily for women holds for between 50% and 60% of the edges. That means that women connect around half of the time with other women.

For men we observe that more than 60% of the connections are to nodes of the other gender, female. The fact that women have more connections among themselves is known by other studies on gender and relationships [7]. However, the figure also shows that homophily is not present for the male-male connection (i.e., new connections of men are more often with women). When taking both categories together, there is homophily on gender: above 60% of the edges connecting people of the same gender.

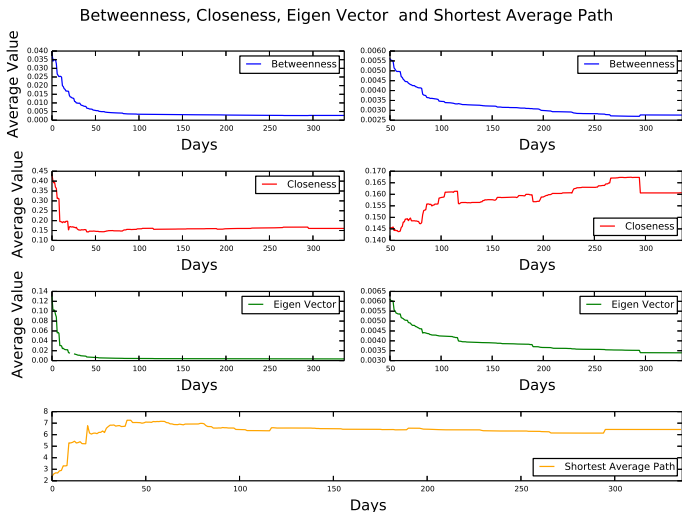


Fig. 5: Mean of all centrality measures for all nodes at each time step (six graphics on top) and average shortest path (bottom)

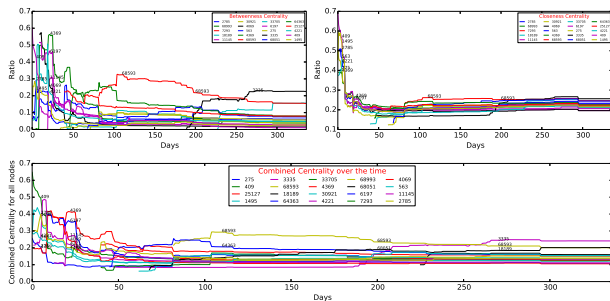


Fig. 6: Centrality for most central nodes. Betweenness (upper left) and closeness (upper right) for the 20 nodes with the highest combined centrality. Lower graphic shows the combined centrality measures

4.5 Identifying influential participants

The dynamic nature of the network is clearly visible from the analyses presented in the previous sections. In previous work we have shown that the more successful participants in the program are, the smaller is the density of their ego-network [10]. This section demonstrates that the set of most influential participants dynamically changes over time. We identify influential participants by comparing properties such as betweenness centrality, closeness centrality, eigenvector centrality, ego-network density and average shortest path.

Figure 8 shows the relation between the node degree of each participant and their ego-network density for the first and last day of the experiment. In this graph we're interested in nodes that have a low density yet a growing degree, as they can be bridges on spreading of emotions, for instance. These are the participants in the top-left quadrant of the graph. Despite the fact that this is just a

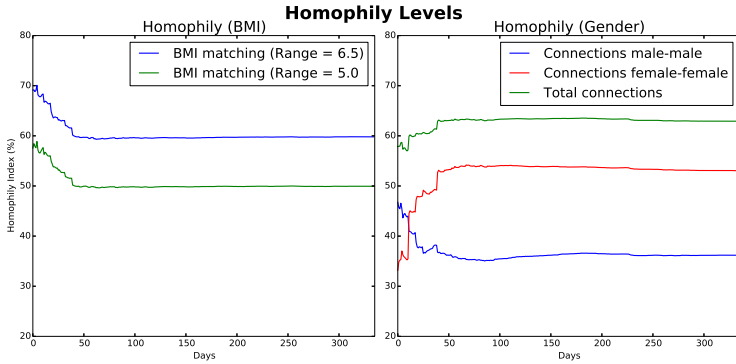


Fig. 7: Homophily according to the BMI (left) and gender (right) of the nodes

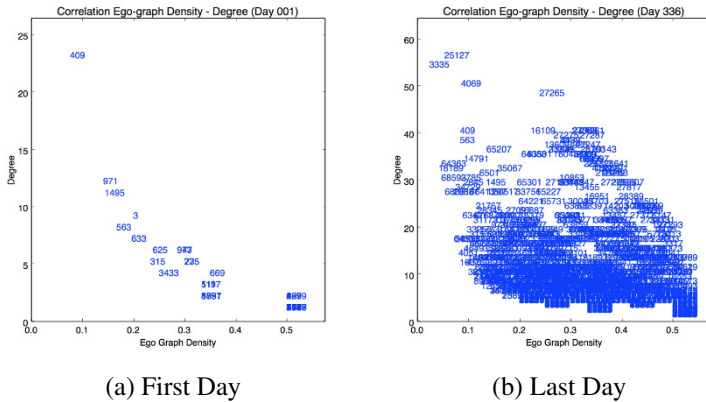


Fig. 8: The dynamic relation between ego network density and nodes’ degrees

snapshot, the changes over time provided by the combination of each day’s relation can give a better picture of what is happening inside a network.

We plotted graphs for all days of the dynamic network which revealed that the set of nodes that emerges in the top-left quadrant are frequently changing. During the experiment, four leader nodes were in evidence considering the ratio between the degree of the nodes and the ego-network density. Node 409 (from day 1 to 12), node 3069 (from day 13 to 40), node 25127 (from day 41 to 254) and node 3335 (from day 255 to 336).

5 Conclusions

In this paper, we have investigated the *dynamic* properties of a longitudinal study of a networked community participating in an online health promotion program. It turned out that studying the dynamics gives additional insights in characteristics of the network. For example, it is shown that the number of components in the network is decreasing while the size of the components is increasing

at the same time. The components themselves follow a Power-law distribution at all time steps: there are a few components with many nodes, and a lot of components with only a few nodes. It is also shown that characteristics like betweenness, closeness, eigen vector and average shortest path at the start of the network are very different from the values after 356 days; however it turned out that already after 50 to 100 days most measurements were relatively stable.

The dynamical data set also allowed us to evaluate whether two well-known phenomena of evolving networks are present: homophily and preferential attachment. Our analysis showed that homophily takes place on the aspect BMI and gender; the latter especially for female-female connections. Apart from the possible preferential attachment, more investigation is needed to affirm that it is present in this data set.

Finally, the combination of degree measurements and the density of the ego-network was presented, and we aim to use it to identify people that are potentially influential in their network in further work. Interestingly, the set of people who are influential according to this metric changes during the evolution of the network, even after the moment that the nodes of network have stabilized. This suggest that continuous monitoring the evolution of a network is important to identify such people.

We believe our discoveries and methods can form the basis for automated (health) interventions that exploit the social network for changing behaviours of individuals, and possibly lead us to future discoveries about leadership, spreading of emotions or any other application related to the network's topology and dynamics.

Acknowledgements E.F.M. Araujo's funding is provide by the Brazilian Science without Borders Program, through the Coordination for the Improvement of Higher Education Personnel, CAPES (reference 13538-13-6).

References

- [1] Acemoglu, D., Ozdaglar, A.: Opinion dynamics and learning in social networks. *Dynamic Games and Applications* **1**(1), 3–49 (2011)
- [2] Acemoglu, D., Ozdaglar, A., ParandehGheibi, A.: Spread of (mis) information in social networks. *Games and Economic Behavior* **70**, 194–227 (2010)
- [3] Araújo, E.F.M., Tran, A.V.T.T., Mollee, J.S., Klein, M.C.A.: Analysis and evaluation of social contagion of physical activity in a group of young adults. In: *ACM International Conference Proceeding Series*, vol. 07-09-Ocob (2015)
- [4] Barabási, A.L., Albert, R.: Emergence of scaling in random networks. *Science* **286**(October), 509–512 (1999)
- [5] Blankendaal, R., Parinussa, S., Treur, J.: A temporal-causal modelling approach to integrated contagion and network change in social networks. In: *Proceedings of the 22nd European Conference on Artificial Intelligence, ECAI16* (2016)
- [6] Christakis, N.a., Fowler, J.H.: The spread of obesity in a large social network over 32 years. *The New England journal of medicine* **357**(4), 370–9 (2007)
- [7] Duck, S., Wright, P.H.: Reexamining gender differences in same-gender friendships: A close look at two kinds of data. *Sex Roles* **28**(11-12), 709–727 (1993)
- [8] Ellison, N.B., Steinfield, C., Lampe, C.: The benefits of facebook "friends": Social capital and college students' use of online social network sites. *Journal of Computer-Mediated Communication* **12**(4), 1143–1168 (2007)
- [9] Eubank, S., Guclu, H., Kumar, V.S., Marathe, M.V., Srinivasan, A., Toroczkai, Z., Wang, N.: Modelling disease outbreaks in realistic urban social networks. *Nature* **429**(6988), 180–184 (2004)

- [10] Groenewegen, M., Stoyanov, D., Deichmann, D., van Halteren, A.: Connecting with active people matters: the influence of an online community on physical activity behavior. In: International Conference on Social Informatics, pp. 96–109. Springer (2012)
- [11] Kempe, D., Kleinberg, J., Tardos, É.: Influential Nodes in a Diffusion Model for Social Networks. *Automata, Languages and Programming* **3580**, 1127–1138 (2005)
- [12] Lazarsfeld, P.F., Merton, R.K.: Friendship as a Social Process: A Substantive and Methodological analysis. *Freedom and Control in Modern Society* **18**, 18–66 (1954)
- [13] Manzoor, A., Mollee, J.S., Araújo, E.F., van Halteren, A.T., Klein, M.C.A.: Online sharing of physical activity: does it accelerate the impact of a health promotion program? In: *Socialcom 2016* (2016)
- [14] Mcpherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology* **27**(1), 415–444 (2001)
- [15] Mislove, A., Marcon, M., Gummadi, K.P., Druschel, P., Bhattacharjee, B.: Measurement and analysis of online social networks. *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement - IMC '07* pp. 29–42 (2007)
- [16] Newman, M.E.J.: The structure and function of complex networks. *Siam Review* **45**(2), 167–256 (2003)
- [17] Newman, M.E.J., Watts, D.J.: Scaling and percolation in the small-world network model. *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics* **60**(6 Pt B), 7332–7342 (1999)
- [18] Organization, W.H., et al.: Global database on body mass index: an interactive surveillance tool for monitoring nutrition transition. World Health Organization: Geneva (2012)
- [19] Scott, J.: *Social Network Analysis*. Sage (2012)
- [20] Tsvetovat, M., Kouznetsov, A.: *Social Network Analysis for Startups: Finding connections on the social web.* " O'Reilly Media, Inc." (2011)
- [21] Valente, T.W.: *Network models of the diffusion of innovations*, vol. 2. Hampton Press (NJ) (1995)
- [22] Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. *Nature* **393**(6684), 440–2 (1998)

Social Networks and Construction of Culture: A Socio-Semantic Analysis of Art Groups

Nikita Basov, Ju-Sung Lee and Artem Antoniuk

Abstract This paper explores the relations between social ties and cultural constructs in small groups. The analysis uses cross-sectional data comprising both social networks within three art groups and semantic networks based on verbal expressions of their members. We examine how positions of actors in the intragroup social networks associate with the properties of cultural constructs they create jointly with other group members accounting for different roles actors play in collective culture constructing. We find that social popularity rather hinders sharing of cultural concepts, while those individuals who socially bridge their groups come to share many concepts with others. Moreover, focusing and, especially, integration of cultural constructs, rather than mere ‘thickness’ of those, accompany intense interactions between the leaders and the followers.

1 Introduction

Network analysts combining culture and networks have shown that culture is linked to social relations. On the one hand, researchers argue that culture is reproduced through interactions and therefore relies on concrete interpersonal ties (e.g., [7, 8, 36]). On the other hand, it is shown that culture affects structure of social ties (e.g., [14, 23]). In sum, culture and social networks are seen as mutually constitutive, or dual [5, 25].

Most of the above studies view culture as a set of constructs which combine ideas, concepts, and meanings shared among individuals (for an overview, see [25]). These constructs correspond to similar ways of interpreting the world and condition similarities in preferences, tastes, ideas, and judgments (for an overview, see [30]). Cultural constructs are exhibited in verbal (written or spoken) expressions of people who belong to the same culture [26, 38]. In these expressions, structures of associations between words rather than the words themselves represent cultural meanings [33, 37]. Hence, research has been advocating a structural view on verbally expressed culture [6, 9]. Yet, the relations between social networks and culture as *structure* have not been sufficiently analyzed, especially in small groups (see in relevant overviews by [25] and [5]). This paper investigates *how*

Nikita Basov (e-mail: n.basov@spbu.ru) and Artem Antoniuk (e-mail: artiom.a000@gmail.com)

Centre for German and European Studies, St. Petersburg State University – Bielefeld University, St. Petersburg, Russia

Ju-Sung Lee (e-mail: lee@eshcc.eur.nl)
Erasmus University Rotterdam, Rotterdam, The Netherlands

social network positions of actors in the social networks associate with cultural constructs they create jointly with other group members.

Using semantic network analysis based on word collocation [6, 16, 32, 34], we trace cultural constructs as patterns of associations between concepts expressed by individuals' and relate the properties of those cultural constructs to positions in networks of social ties occupied by individuals. Hence, we apply the growingly popular socio-semantic framework [27, 29, 31, 32].

In particular, we focus on groups of visual artists. These groups jointly generate culture, most often in observable processes of creating corporeal artistic objects and group interactions, exchanging on – often joint – artwork creation, collective exhibitions, discussions on the events and figures of the artistic scenes, and other artistic and everyday topics. Network analysis has been widely applied to study creativity and social relations between artists. Yet, network studies of art have focused primarily on organizational and market levels (e.g., [3, 13, 15, 35]), while creativity is seen as dependent on an individual's [11, 21] position in a network of *external* relations [20]. The question of how internal networks of art groups operate appears to be out of scope. Meanwhile, it is those internal networks of art groups that bring to life novel artistic visions and artistic styles many groups strive for [17] thus generating variations in culture. So, it makes sense to take a closer look at such internal social networks. Simultaneously, research on language use has been argued to be “a powerful way to study the collective action of cultural production in art worlds” [12, p. 201]. Developments in semantic networks allow for the exploration of relations between cultural production and social networks within art groups. Yet, so far, very few studies applied formal semantic network analysis techniques to artistic settings [1]. This paper deals with this gap.

2 Data

The empirical data used in this study covers 3 art groups from St. Petersburg, Russia, encoded as ‘A’, ‘B’, and ‘C’. All of them are working in the format of contemporary visual art. They all are characterized by intense interaction between the members, (decades-long) backgrounds shared by most of the members, and regular joint artistic and/or everyday practices. Hence, their cultural constructs may both affect their interactions and be impacted by these. Besides, the groups actively produce texts and narratives that can be used to capture expressed cultural constructs. Simultaneously, the groups are different in organization, educational and cultural backgrounds of their members, understandings of art and its tasks, forms of spatial embeddedness in the city space, and artistic styles. This provides variability in cases.

We collected data between 2011 and 2012 via in-depth ethnographic studies conducted in each of the 3 groups. Because the groups do not have formal boundaries, we decided to include only core members in the data collection, that is those members with stable membership and continuous involvement in the group practice.

The data consists of two main parts: textual data and sociometric data. The textual data includes verbal expressions of the group members with clearly identifiable individual authorship. The corpus of texts is composed of transcripts of 24 open-ended narrative interviews, each 30–240 minutes long, transcripts of dialogues between group members coming from 17 ethnographic observations, each 2–8 hours long, as well as posts in Russian social media, textual works of the artists, such as newspaper articles, prose and poetry. We managed to gain texts by every core member in all the 3 groups. Unprocessed individual corpora sized between 4128 and 28928 words per member.

The sociometric data was obtained using the roster recall method surveys capturing frequency of interactions among members of each group. The question asked was “How often do you interact?”, suggesting to choose from 5 response options to evaluate frequency of interactions with each of other members of the group: almost never; 1 or less/month; 2–4 times/month; 5–14 times/month; 15 or more times/month. Further, responses were quantified on ordinal levels, from 0 for ‘almost never’ to 4 for ‘15 or more times/month’. 25 out of 29 core members responded to the survey resulting in a response rate of 86.21%.

3 Method

3.1 Mapping of the Socio-Semantic Networks

To capture patterns of social ties, cultural structures, and structure of relations between them, three types of networks were mapped using the data on the three art groups: actor-actor (social network representing structure of social ties), concept-concept (semantic network representing cultural constructs) and actor-concept (bimodal concept usage network representing links between individuals and certain cultural constructs). Combined, these three types of networks constitute socio-semantic networks [32].

The edge widths of the actor-actor (social) networks in Fig. 1 are based on ordinal levels from 0 to 4 captured by the sociometric survey. Tie strength was taken as an average of individuals' evaluations of frequency of interactions with each other. When no response was received from one of the individuals in a dyad, only the strength indicated by the other one served as an input for the social network.

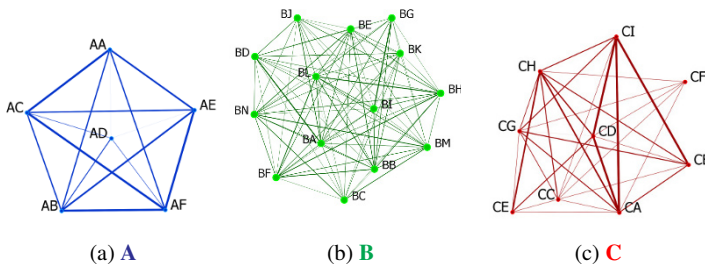


Fig. 1: Social networks of the 3 art groups: “A”, “B”, and “C”.

The other two types of networks were mapped based on the collected texts. Concepts, which are stems of words used in texts, are the nodes in semantic and actor-concept networks. To map relations between concepts in semantic networks we used words collocation technique, which implies that links between nodes are mapped based on word stems co-occurrences in texts. These networks represent cultural constructs expressed by individuals [6, 25]. Relations between concepts and actors were mapped based on usage of certain concepts by certain individuals in their texts. Neither frequency of words' collocation nor frequency of words use were accounted for in this analysis, so both semantic and concept usage networks are binary.

The procedure for mapping semantic and concept usage networks was as follows. First, the textual data were split into separate files containing all narratives and written texts by each single group member, separately. Then, we removed interviewers' and observers' comments and technical information. Second, textual data were preprocessed in AutoMap [10], applying concept stemming, lowercasing, removal of punctuation and numerals. A delete list was created and applied, removing pronouns, adverbs, prepositions, conjunctions, junk words, as well as less meaningful verbs, such as 'say', 'talk', and 'think'.

Third, AutoMap was applied to each of these separate files to generate *individual semantic networks* of each artist. Parameters of semantic network generation were specified as follows: window size between 2 and 3 words was used to map lines between concepts; sentence was used as a stop unit.

Fourth, individual networks of each group member were aggregated into *union semantic networks* (so that links are now based on collocation of concepts in texts of any of the artists), while actor-concept networks still contained the information on usage of certain concepts by certain actors.

Fifth, concepts used by only one group member (i.e. having fewer than 2 binary actor-concept links) were removed from semantic and actor-concept networks as we are interested only in capturing shared cultural constructs. Therefore, our analysis includes only concepts used by at least two actors in a group. We note, however that in this paper, links between concepts are not necessarily shared.

The three types of networks (social, semantic, and bipartite concept usage) were mapped for each of the three art groups, resulting in 12 networks in total and comprising 3 socio-semantic networks of the 3 groups further used in this analysis.

3.2 Operationalization of the Social Network

The social network survey recorded the frequency of interactions among members of each group. However, this frequency was measured on ordinal levels, which carry concerns over numerical comparisons from one level to the next. For example, the ratios among levels differ from any estimated levels. So, we instead replace tie strengths with estimations of the actual frequency of contact.

Table 1 enumerates the estimates and ranges (for sampling) for tie strength ordinal scale values. In our subsequent analyses, the estimates, rather than the survey responses, are employed. An

Table 1: Mapping of Tie Strengths to Ranges and Estimates

Survey		Min.	Max.	Estimate
Response	Description			
0	Almost Never	0.01	0.1	0.05
1	1 or less/month	0.1	1.0	0.5
2	2–4 times/month	1.5	4.5	3.0
3	5–14 times/month	4.5	14.5	9.5
4	15 or more times/month	14.5	20.0	20.0

alternative approach is to consider uniform sampling of tie strengths using the estimated min and max actual frequencies (also shown in the tables). Finally, asymmetric interaction reports are symmetrized by averaging the dyadic reports.

3.3 Descriptive Statistics

We present some descriptive statistics of the social and semantic networks in Table 2. These groups are relatively small and may be considered to be small social networks, which bear the characteristic of being socially cohesive. That is, the social networks are highly dense. By contrast, the semantic networks exhibit extremely low densities, which is largely due to the high number of concept nodes and the co-word window employed in the semantic network generation. Despite the huge difference between the densities of social and semantic networks, we note that these densities are ordered similarly, suggesting a relation between social networks and cultural constructs. For example, the “C” network exhibits both the lowest concept and actor network densities.

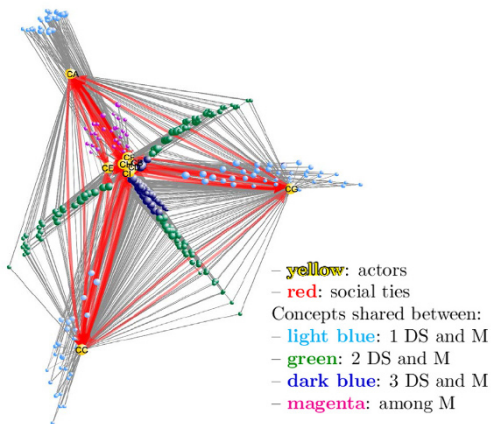
Table 2: Social and semantic network statistics

	A	B	C
Actors	6	14	9
Ties	15	89	28
Ord. Weighted Ties	44.5	152	53
Est. Wgt. Ties	163.75	284	141.75
Interactions/Tie	10.92	3.19	5.06
Social Network Density	1.000	0.978	0.778
Concepts	7513	4800	13681
Semantic Network Density	0.00077	0.00058	0.00039

3.4 Extraction of socio-semantic subgraphs

In Fig. 2, we visualize the union graph of the bipartite concept usage network and the unimodal social network for the ‘C’ group. The concept usage network is optimized using the pivot multidimensional scaling (MDS) algorithm [4], as implemented in Pajek [2], so that structural equivalence is optimally displayed. That is, nodes that are connected to similar others are placed in proximity to one another and nodes connected to the same other nodes – exactly upon each other, thereby reducing the visual complexity induced by the 13,681 observed concepts. Clusters of concepts form distinguishable groups or ‘bands’ of concept nodes scattered around nodes of actors using them (See Fig. 3). The added value of such an optimization is that it gives a picture of how actors are grouped together with regard to usage of similar sets of concepts and concepts are grouped with regard to their usage by certain sets of actors.

Fig. 2 Visualization of actor-concept and actor-actor networks of group ‘C’.



Actor nodes are in yellow and labeled (anonymously). Concept nodes are colored according to combinations of actors sharing them and sized by the number of structurally-equivalent concepts. Grey lines refer to concept usages, while overlaid red lines represent social ties.

Due to the nature of pivot MDS algorithm, some actor nodes usually form a triangle-shaped structure with other actor nodes located in the middle of the diagram. Actor nodes located at

triangle's vertices represent people who use the largest amount of concepts shared with other members, while actor nodes located inside represent those who use a significantly smaller number of shared concepts. Thus, a distinction between different positions of actors in the concept usage network is captured. Actor nodes located a triangle's vertices appear to be very different in their cultural constructs with regard to each other, as reflected by the content of the concepts they use. What they have in common is that they use many concepts, which are also used by many others in the group. Hence, they span the semantic space playing an important role in culture constructing in the group. Therefore, we label them 'discourse spanner(s)' (or **DS**). Simultaneously, these individuals appear to be informal leaders in their groups, acknowledged as such by other members and demonstrating corresponding behavior in group interactions. Due to their strong involvement in the formation of their groups' shared semantics, characteristics of the semantic networks that DS contribute to are most worth considering in order to understand how culture is constructed in groups.

The second type of position in the socio-semantic network is represented by the 'majority' (or **M**) of other actors who are using shared concepts to a much lesser extent and hence are less involved in culture constructing within their groups. The positions of DS and M represent not only two different types of positions in the concept usage structure, but also two distinctive roles in group culture constructing corresponding to these positions. Although in this paper we mainly focus on the DS, we still account for M.

As an important technical step, for each group we extracted different socio-semantic subgraphs. These subgraphs include (1) different combinations of DS and M actors, (2) concepts shared by them, as suggested by the pivot MDS optimization (e.g., blue concepts in Fig. 2 correspond to concepts shared by 1 of the DS and one or more of the M), and (3) any links between concepts they have in their semantic networks; we note that links between concepts are not necessarily shared.

Fig. 3 represents an example semantic network of a socio-semantic subgraph which includes concepts used by the DSs, encoded as CC and CG, share with some of the M (green nodes in Fig. 2). It represents, for instance, that the concepts 'poem' and 'prose' are used by DS 'CC', DS 'CG' and at least one individual in M; meanwhile, the association represented by the link 'poem'-'prose' may be characteristic of only CC, or CG, or the individual(s) from among the M.

Due to the limits of space, our analysis in this paper considers only those socio-semantic subgraphs that include one DS and one or more of the M.

4 Results

As a starting point, in Table 3, we predict (in the statistical, non-causal sense) concept usage by individual actors' social network position statistics, namely degree (C_D) and betweenness (C_B) centralities.² The former measure captures the extent to which an actor interacts with others, while the latter indicates the extent to which an actor plays a bridging role within his/her group. [18, 19]. We examine centrality measures derived from the undirected, unweighted graphs as well as the estimated, empirical edge weights, in order to address homogeneity of unweighted degree centralities due to high density in some groups. The models are applied across all groups (total of 29 members), and the dependent variable is log-transformed due its skewness.

The results primarily reveal that betweenness (C_B) is positively associated to shared concept usage by individuals while degree (or popularity, C_D) is negatively associated. The relative, absolute magnitudes of these effects vary by the operationalization of the tie, whether it is mere existence or

¹ Node size corresponds to betweenness centrality of concepts. Pendants were recursively hidden in the main picture. The full semantic network is displayed in the lower-right.

² Due to the low sample size, we cannot include additional predictors or employ a nested model. However, group size, while significant on its own, is collinear with C_D , but does not predict as well not do group-level dummy variables.

the strength.³ That is, actors use and share with others more concepts when they connect areas of their social networks, while they use and share fewer concepts when they intensely interact with their closer circles. This reveals that conceptual prominence of DS is hindered by their popularity but is empowered by their ability to connect the group. Given that degree and betweenness are often positively correlated, the negative effect of C_D reveals that those DS who use particularly many concepts are rather distinctive gatekeepers than merely globally central through high ranking on both measures [27, 28, 29].

In Table 4, we compare the weighted social network to the ‘concept sharing network’; the latter is the bipartite concept usage network transformed via network multiplication (or folding) into a unimodal actor-actor network in which the edge weight represents the extent of shared concepts. For each group and socio-semantic subgraph corresponding to different types of roles (DS and M), the correlations between the edges of the social network and those of the concept sharing network are tested for significance under a permutation test that produces a null distribution resistant to type I errors induced by matrix (network or distance) auto-correlation [24]; the resulting correlation is called a ‘QAP correlation’ [22]. We examine the relationship between the social ties and concept sharing by pairs of actors for each group, in general, as well as for DS and M subgraphs within each group. These subgraphs strictly contain only social ties among DS (or M, respectively) and the concepts DS (or M) share between them. The edge weights in the concept sharing networks are additionally log-transformed due to skewness.

Table 4: QAP correlations between shared concepts and social ties per subgroup

Name	Pearson r	Pearson r (w/log trans.)	n
<i>Discourse Spanners (DS) Subgraph</i>			
A	—	—	2
B	.298 ^{n.s.}	.296 ^{n.s.}	4
C	.345 ^{n.s.}	.399 ^{n.s.}	3
<i>Majority (M) Subgraph</i>			
A	.041 ^{n.s.}	.035 ^{n.s.}	4
B	.123 ^{n.s.}	.128 ^{n.s.}	10
C	-.337 [^]	-.401 [^]	6
<i>All Members Graph</i>			
A	-.034 ^{n.s.}	-.001 ^{n.s.}	6
B	.034 ^{n.s.}	.105 ^{n.s.}	14
C	-.284 [^]	-.350 [*]	9
<i>n.s. : $p \geq .10$; [^] : $p < .10$; * : $p < .05$</i>			

Despite the small samples, there are some results worth mentioning. First, we see that the DS social and concept sharing networks (for those groups that contain more than two DS) exhibit positive, albeit insignificant, correlations. These suggest the social ties between DS as a subgroup and concept sharing between them have an ambivalent association: either strong ties act as a normalizing force on inducing a common dictionary or vice versa.

³ Results from considering weighted concept usage (multiple use per concept by a single individual) are very consistent with the shown results.

The M subgraphs also exhibit this ambivalence with the exception of group C, whose significantly negative correlation indicates that the more strongly M actors are tied, the fewer concepts they share. This suggests that certain M members of group C sought distinction from one another in their cultural constructs. This correlation remains when we look at the entire C group despite the normalizing nature of the DS of that group.

As the above analysis shows, mere concept sharing by individuals does not demonstrate any prominent relation with social ties. Further, we account for cultural meanings by considering links between concepts (semantic networks), and we search for relations between social ties linking actors and cultural constructs the actors jointly express. We compare semantic network statistics of subgraphs that include one DS and the M (i.e. separate union semantic networks connecting concepts shared by DSs and one or more of Ms) against normalized sum of weights (i.e. the sum of dyadic degree centralities divided by maximum sum possible) of the edges a DS has with the M. Specifically, we compare graph-level measures (GLMs) computed for the semantic networks (per sets of DS+M) against the interaction strength the DS exhibits with the M in their respective groups. This comparison (Table 5) exposes the extent to which the cultural constructs created by discourse spanners together with the majority of other actors in their groups relate to cumulative strength of social ties between the DS and the M within a group.

Table 5: Semantic network statistics v. averaged social network tie strengths

Measure	r_{ord}	r_{est}	\bar{r}_{MC}
Density	.116 ^{n.s.}	-.047 ^{n.s.}	-.001 ^{n.s.}
Degree Centralization	.375 ^{n.s.}	.259 ^{n.s.}	.314 ^{n.s.}
Betweenness Centralization	.883**	.822**	.814*

n.s. : $p \geq .10$, * : $p < .05$, ** : $p < .01$

There are $n = 9$ DS (3, 2, and 4 for each group respectively). Density indicates the unweighted density of the semantic networks in each DS+M socio-semantic subgraph. Degree and betweenness centralizations are variance-based metrics of the distribution of nodal degree and betweenness centralities and are normalized between 0 and 1. They reveal the extent to which the structure contains concepts that a) harbor significantly more semantic linkages to other concepts and b) play prominent bridging roles in the semantic network, connecting disparate areas of a group’s cultural constructs and thus integrating them. Together, they can describe properties of cultural constructs. For example, higher density would suggest ‘thickness’ of cultural constructs; and lower degree centralization may indicate more diversified cultural constructs, in contrast to those that are narrowly focused.

We report the nominal Pearson correlations (r) derived from both the ordinal responses and empirical estimates as well as a mean from Monte Carlo sampled tie strengths; these are all consistent with one another for the higher correlations. Significant and positive correlations for betweenness centralization indicate that the presence of distinct concepts that prominently bridge semantic networks accompanies stronger bonds between a DS and a M. In other words, strong social ties are associated with integration of cultural constructs. The other positive correlation, for degree centralization, although insignificant, points towards decreased diversification of cultural constructs as being associated with stronger social ties between a DS and a M. It suggests that the cumulatively strong ties between DS and M may make group discourse elaborate on some narrow set of focal concepts, perhaps mobilizing the group discourse. Thus, focusing and, especially, integration of cultural constructs rather than mere ‘thickness’ of cultural constructs accompany intense interactions between DS and the M.

5 Conclusion

This study focused on relating social networks and cultural constructs in art groups, with implications on social and cultural duality extending to other domains. By studying the interplay between social and semantic networks, we attempted to shed light on the relation social role and position of an individual have with his/her involvement in constructing shared culture in a group. Minding that our findings are limited due to analysis of cross-sectional data on small groups embedded in a single – artistic – setting, we can summarize the following.

First, the analysis revealed that, even in small groups of friends, higher diversity and intensity of direct social ties hinders sharing of cultural constructs. Rather, those individuals, who socially bridge less well-connected areas of their groups, are the ones who engage in the shared cultural constructs with others.

Second, the amounts of concepts shared by the group members and strength of social ties between them are not necessarily related. While those individuals using significant amounts of shared concepts bear some of this association (which Roth and Cointet refer to as “semantic homophily” [32], in one of the groups, the members employing significantly fewer shared concepts exhibit heterophily, whereby stronger ties are marked by lower levels of concept sharing. This finding differs from those of [32] (*ibid.*) which, however, rely on analysis of much larger groups.

Finally, we found that stronger focusing and higher integration of cultural constructs rather than mere ‘thickness’ of cultural constructs accompany more intense interactions between the leaders and the followers. Our preliminary interpretation is that leaders are strategically interacting with others in order to jointly construct a shared creative vision and to integrate the group. In this process, leaders rely not only on their competence or formal authority, but also on focusing on emerging cultural constructs and on interaction with others. The more intensely they interact with the rest of the group, the more they bridge and focus the individual group’s cultural constructs on a shared set of concepts serving to span the group’s culture. At the moment, we cannot say for sure whether or not it is a phenomenon specific to creative settings, and if there is an asymmetric relation. This issue will be addressed in our analysis of longitudinal data, currently being collected.

Overall, we can preliminarily conclude that the socio-semantic network approach is capable of delivering findings on the duality of cultural and social structures relevant to the ongoing discussion (see [5, 25] Yet, the analysis would benefit from a more extensive account of links between concepts (semantic networks) and from combining of quantitative and qualitative data. We expect that joint formal analysis of semantic network properties with contents of semantic networks, along with ethnographic and interview data, will deliver further insights.

Authors’ Contributions The first two co-authors contributed equally to this paper and are listed alphabetically.

Acknowledgements The paper has benefited from the support of: St. Petersburg State University (“Communication practices of knowledge creation in the social space of a contemporary city”, 2011-2012), Russian scientific foundation for humanities (15-03-00722 “Coevolution of knowledge and communication networks: structural dynamics of creative collectives in European cultural capitals”, 2015–ongoing), and the Centre for German and European Studies Bielefeld University and St. Petersburg State University supported by the DAAD with funds from the German Foreign Office. Also, the authors express their gratitude to those who helped in data collection and processing: Aleksandra Nenko, Anisya Khokhlova, Elena Tykanova, Maria Veits, Olga Volkova, Irina Shirobokova, Alexey Evstifeev, Alexander Kopyi, and Olga Nikiforova. We would also like to thank Wouter de Nooy, who proposed to use pivot MDS optimization for the given data, and Adina Nerghes for her comments and logistical assistance in assembling this paper. We are also very grateful for comments received on this paper from Iina Hellsten.

References

- [1] Basov, N., Nenko, A.: Artistic community knowledge structure revealed: A semantic network analysis of ‘La Escocesa’, Barcelona. Centre for German and European Studies pp. 3–32 (2013)
- [2] Batagelj, V., Mrvar, A.: Pajek-program for large network analysis. *Connections* **21**(2), 47–57 (1998)
- [3] Bottero, W., Crossley, N.: Worlds, fields and networks: Becker, Bourdieu and the structures of social relations. *Cultural Sociology* **5**(1), 99–119 (2011)
- [4] Brandes, U., Pich, C.: An experimental study on distance-based graph drawing. In: 16th International Symposium on Graph Drawing, vol. 5417, pp. 218–229. Springer-Verlag (2008)
- [5] Breiger, R.L., Puetz, K.: Culture and networks. In: International encyclopedia of social and behavioral sciences (2nd. ed ed.). Elsevier, New York (2015)
- [6] Carley, K.: Extracting culture through textual analysis. *Poetics* **22**(4), 291–312 (1994)
- [7] Carley, K.M.: An approach for relating social structure to cognitive structure. *Journal of Mathematical Sociology* **12**(2), 137–189 (1986)
- [8] Carley, K.M.: A theory of group stability. *American Sociological Review* **56**(3), 331–354 (1991)
- [9] Carley, K.M.: Extracting team mental models through textual analysis. *Journal of Organizational Behavior* **18**(1), 533–558 (1997)
- [10] Carley, K.M., Columbus, D., Landwehr, P.: Automap User’s Guide 2013. Tech. rep., CMU-ISR-CASOS, Pittsburgh, PA (2013)
- [11] Cattani, G., Ferriani, S.: A core/periphery perspective on individual creative performance: Social networks and cinematic achievements in the Hollywood film industry. *Organization Science* **19**(6), 824–844 (2008)
- [12] Cluley, R.: Art words and art worlds: The methodological importance of language use in Howard S. Becker’s “Sociology of art and cultural production”. *Cultural Sociology* **6**(2), 201–216 (2012)
- [13] Crane, D.: The transformation of the avant-garde: The New York art world, 1940–1985. University of Chicago Press, Chicago, IL (1989)
- [14] Dahlander, L., McFarland, D.A.: Ties that last: Tie formation and persistence in research collaborations over time. *Administrative Science Quarterly* **58**(1), 69–110 (2013)
- [15] De Nooy, W.: A literary playground: Literary criticism and balance theory. *Poetics* **26**(5), 385–404 (1999)
- [16] Diesner, J.: From texts to networks: Detecting and managing the impact of methodological choices for extracting network data from text data. *KI-Knstliche Intelligenz* **27**(1), 75–78 (2013)
- [17] Farrell, M.P.: Collaborative circles: Friendship dynamics and creative work. University of Chicago Press, Chicago, IL (2003)
- [18] Freeman, L.C.: A set of measures of centrality based on betweenness. *Sociometry* **40**(1), 35–41 (1977)
- [19] Freeman, L.C.: Centrality in social networks: Conceptual clarification. *Social Networks* **1**(3), 215–239 (1979)
- [20] Guimera, R., Uzzi, B., Spiro, J., Amaral, L.A.N.: Team assembly mechanisms determine collaboration network structure and team performance. *Science* **308**(5722), 697–702 (2005)
- [21] Hargadon, A., Sutton, R.I.: Technology brokering and innovation in a product development firm. *Administrative Science Quarterly* **42**, 716–749 (1997)
- [22] Krackhardt, D.: QAP partialling as a test of spuriousness. *Social Networks* **9**, 171–186 (1987)
- [23] Lizardo, O.: How cultural tastes shape personal networks. *American Sociological Review* **71**(5), 778–807 (2006)
- [24] Mantel, N.: The detection of disease clustering and a generalized regression approach. *Cancer Research* **27**(2), 209–220 (1967)
- [25] Mohr, J.W.: Measuring meaning structures. *Annual Review of Sociology* **24**, 345–370 (1998)

- [26] Mohr, J.W., Duquenne, V.: The duality of culture and practice: Poverty relief in New York City, 1888–1917. *Theory and Society* **26**(2), 305–356 (1997)
- [27] Nerghes, A., Hellsten, I., Groenewegen, P.: A toxic crisis: Metaphorizing the financial crisis. *International Journal of Communication* **9**, 106–132 (2015)
- [28] Nerghes, A., Lee, J.S., Groenewegen, P., Hellsten, I.: The shifting discourse of the European Central Bank: Exploring structural space in semantic networks. In: K. Yetongnon, A. Dipanda (eds.) *Proceedings of the 10th SITIS*, pp. 447–455. IEEE Computer Society (2014)
- [29] Nerghes, A., Lee, J.S., Groenewegen, P., Hellsten, I.: Mapping discursive dynamics of the financial crisis: A structural perspective of concept roles in semantic networks. *Computational Social Networks* **2**(16) (2015)
- [30] Pachucki, M.A., Breiger, R.L.: Cultural holes: Beyond relationality in social networks and culture. *Annual Review of Sociology* **36**(1), 205–224 (2010)
- [31] Roth, C.: Socio-semantic frameworks. *Advances in Complex Systems* **16**, 0405 (2013)
- [32] Roth, C., Cointet, J.P.: Social and semantic coevolution in knowledge networks. *Social Networks* **32**(1), 16 – 29 (2010)
- [33] Saussure, F.D., Hidayat, R.S.: *Pengantar Linguistik Umum*. Gajah Mada University Press, Yogyakarta, Indonesia (1988)
- [34] Sinclair, J.: *Corpus, concordance, collocation*. Oxford University Press, Oxford (1991)
- [35] Uzzi, B., Spiro, J.: Collaboration and creativity: The small world problem. *American journal of sociology* **111**(2), 447–504 (2005)
- [36] White, H.C.: *Identity and control: A structural theory of social action*. Princeton University Press, Princeton, NJ (1992)
- [37] Wittgenstein, L.: *Philosophical Investigations* (Vol. 50). Prentice Hall, New York (1953)
- [38] Yeung, K.T.: What does love mean? Exploring network culture in two network settings. *Social Forces* **84**(1), 391–420 (2005)

Water Supply Network Partitioning Based On Weighted Spectral Clustering

Armando Di Nardo, Michele Di Natale, Carlo Giudicianni, Roberto Greco and Giovanni Francesco Santonastaso

Abstract Water Network Partitioning (WNP) in District Meter Area (DMA), obtained inserting remote control valves and flow meters in water supply systems, allows simplifying the water balance and pressure control in order to reduce water leakage and to improve water quality protection. Traditionally, the WNP is based on empirical suggestions and on trial and error approaches used with hydraulic simulation software, difficult to apply to large networks. Recently, some heuristic procedures, based on graph and network theory, have shown that it is possible to find optimal solutions in terms of number, shape and dimension of DMAs. In this paper, spectral clustering theory was used to define the water districts, taking into account the spatial and hydraulic constraints, through weight matrices. A comparison between different spectral clustering methods was achieved on a real water network measuring some energy performance indices, in order to identify the optimal water network partitioning.

1 Introduction

Water Supply Networks (WSNs) are among the most important civil networks, because they deliver drinking and industrial water to metropolitan areas.

From a topological perspective, a WSN can be considered as a planar weighted graph, with n nodes and m links. A WSN with multiple interconnected elements may be represented as a link-node planar spatially organized graph for which pipes (and valves) correspond to links, and nodes/junctions (such as pipe intersections, water sources and nodal water demand) correspond to graph nodes. It belongs to the class of networks with nodes occupying precise positions in two or three-dimensional Euclidean space, edges being real physical connections, and strongly constrained by their geographical embedding [3], like other spatially organized urban infrastructure systems [4, 23].

In an abstract modeling context, a mathematical graph can be used to express the relationships between groups of linked nodes. An important aspect of spatial networks is that node degrees are

A. Di Nardo (e-mail: armando.dinardo@unina2.it) · M. Di Natale (e-mail: michele.dinatale@unina2.it) · R. Greco (e-mail: roberto.greco@unina2.it) · G.F. Santonastaso (e-mail: giovannifrancesco.santonastaso@unina2.it)

Second University of Naples and Action Group CTRL+SWAN of European Innovative Partnership on Water

C. Giudicianni (e-mail: carlo.giudicianni@unina2.it)
Second University of Naples

constrained, since the number of possible connections to a single node is limited by physical space. Furthermore, in a spatial network, it is unlikely to find connections between very distant nodes, due to the distance-dependent cost of the edges and to obvious physical constraints, which determine important limitations to the small-world behavior of the networks [3]. In particular, little variability is observed in the connectivity patterns of the nodes in WSN, no hubs (nodes with much more connections than the others) are present, and most of the nodes have very low degree (two or three and mostly less than five), so in general they present a homogeneous degree distribution [7]. Further, such networks are also equally sensitive to random or malicious failures [2].

WSN can be considered as complex networks for many reasons [21]: they are often very large (up to million nodes and links); they are buried underground, and thus are not easily accessible for monitoring and maintenance; they are strongly looped; their modeling includes equations requiring sophisticated numerical resolution methods; they often present severe water losses. Compared to other civil networks (e.g. gas, electricity, transport, telephone, Internet), some of these WSN characteristics are peculiar, and make their management arduous, with many operational problems (such as water and energy losses).

The implementation of the paradigm of divide and conquer in a WSN [22] allows significantly simplifying the management, defining a Water Network Partitioning that consists in defining some District Meter Areas (DMAs) by inserting gate valves and flow meters along network pipes. WNP represents an important innovation in the management of water supply systems, as it allows improving water losses identification [20], controlling district pressure [1], and protecting users from accidental and intentional contamination [9]. By dividing the water network in DMAs, implementing innovative ICT remote-controlled devices and big data analysis, it is possible to change the traditional approach to management of WSN, transforming the water systems into modern Smart Water Network (SWAN) [11], considered as part of Smart Cities.

WNP has to be obtained in compliance with two major constraints: 1) network connectivity, i.e. each demand node of the water network must be connected to at least one water source, and 2) nodal minimum pressure, each node must have a pressure equal or higher than the minimum level of service that allows satisfying water demand of the users. Therefore, the design of a WNP is a complex challenge for operators, because the permanent partitioning changes the original topological layout of water systems. Indeed, network partitioning, achieved by pipe closures, reduces the overall pipe section availability, with the consequent decrease of network water pressure, especially during peak hours, worsening the level of service offered to users.

In the last decade, different procedures have been proposed in the literature for finding an optimal WNP layout (a review is given in [11, 24]). They are essentially arranged in two different phases: a) clustering, aimed to define the shape and the dimension of the network subsets, based on graph theory algorithms [5, 10, 11, 16, 28], spectral approach [18], multi-agent approach [8, 19], community structure [7, 12]; and b) dividing, aimed to physical partitioning of the network, by selecting pipes for the insertion of flow meters or gate valves, based on iterative [12, 14] or genetic algorithms [11], with the objective to define the optimal layout that minimises the economic investment and the hydraulic performance deterioration.

This paper aims to investigate the feasibility of adopting weighted spectral clustering to design DMAs, comparing different weight combinations, in order to find the best one that allows minimizing hydraulic performance deterioration.

2 Methodology

As described above for previous approaches, the proposed partitioning procedure consists of two phases:

Phase 1, Water network clustering. As known, considering a simple graph $G = (V, E)$, where V is the set of n vertices v_i (or nodes) and E is the set of m edges e_i (or links), a k -way graph clustering problem consists in partitioning V vertices of G into k subsets, P_1, P_2, \dots, P_k such that: $\bigcup_i^k P_i = V$ (the union of all clusters P_i must contain all the vertices V_i), $P_i \cap P_j = \emptyset$ (each vertex can belong to only one cluster P_i), $\subset P_i \subset V$ (at least one vertex must belong to a cluster and no cluster can contain all

vertices) and $1 < k < n$ (the number k of clusters must be different from one and from the number n of vertices). Clustering is usually defined in terms of weighted, undirected graphs, where weights correspond to either similarity scores or distances. So, vertices and edges have associated weights, respectively indicated with $\bar{w}_i > 0$ with $i \in V$, and $\epsilon_{ij} > 0 \in E$ and $\epsilon_{ij} = 0$ if $ij \notin E$.

The graph clustering can be achieved with many procedures finalized to define the optimal layout of each cluster, finding community structures minimizing or maximizing an objective function that emphasizes one of the clustering aims. In literature (a wide review is provided in [3]), several procedures were proposed: k -means; Markov cluster algorithm; spectral methods (coupled with cut-methods, such as min-cut, ratio-cut, normalized-cut); hierarchical clustering; modularity; multi-level-recursive algorithm, and some other methods as, for example, relaxing normalized-cut.

In this paper, the clustering phase to define DMAs for WNP was achieved with a recursive weighted spectral clustering technique, based on a generalized eigenvalue problem using the *average cut* ($Acut$) and the *normalized cut* ($Ncut$) formulations [26]. In particular, different weights were used for the pipes (namely, no-weights, pipe diameter and pipe length) to investigate which of them provide best results.

As known, the list of eigenvalues (together with their multiplicities) is defined as the spectrum of the adjacency $n \times n$ matrix $\mathbf{A}_G = (a_{ij})$ of graph $G = (V, E)$, where $a_{ij} = 1$ if there is a link between nodes i and j and $a_{ij} = 0$ otherwise.

Other formulation computes the list of the eigenvalues of the Laplacian matrix (or Kirchhoff matrix) defined as a $n \times n$ matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}_G$ where $\mathbf{D} = \text{diag}(d_i)$ and d_i is the degree of a node i . A particular Laplacian expression is the normalized Laplacian, defined as $\mathbf{L} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2} = \mathbf{D}^{-1/2} (\mathbf{D} - \mathbf{A}_G) \mathbf{D}^{-1/2}$. As well known, the *average cut* criterion is based on the Laplacian spectrum, while the *normalized cut* formulation is based on the normalized Laplacian [26, 30].

The authors propose an algorithm to define, comparing two spectral clustering criteria, the district meter areas in a WSN, identifying in both cases which weight lead to the optimal DMA design. In this regard, the adjacency matrix will be replaced by weight matrix $\mathbf{W}_{NW} = \mathbf{A}_G$ (in the case with No Weights), \mathbf{W}_D (with weights equal to pipe diameters) and \mathbf{W}_L (with weights equal to pipe lengths) to calculate the corresponding Laplacian matrix and so the corresponding spectrum.

Specifically, the clustering phase for the proposed water network partitioning consists of the following steps:

1. abstraction of the water supply network as a graph $G = (V, E)$;
2. definition of adjacency matrix \mathbf{A}_G and pipes weight matrices \mathbf{W}_{NW} , \mathbf{W}_D , \mathbf{W}_L with three different weight combinations (no-weight, pipes diameter and pipes length);
3. computation of the spectrum of Laplacian matrices \mathbf{L}_{NW} , \mathbf{L}_D and \mathbf{L}_L and normalized Laplacian matrices \mathbf{LN}_{NW} , \mathbf{LN}_D and \mathbf{LN}_L for all weight matrices \mathbf{W}_{NW} , \mathbf{W}_D , \mathbf{W}_L ;
4. computation of the eigenvector \mathbf{v}_2 corresponding to the second smallest eigenvalue λ_2 for each spectrum and ;
5. ordering the vertices, in increasing order, according to their $x_{i,2}$ eigenvector \mathbf{v}_2 components value and then dividing them into two groups by the sign of the component for the graph bipartition [3, 15, 26];
6. check the continuity of the obtained clusters;
7. recursive bipartition of the current sub-graph when $k > 2$;
8. definition of the set of edge-cuts (or boundary pipes) N_{ec} .

Phase 2, Water network dividing. Once obtained the set N_{ec} of the edge-cut (or *boundary pipes*), it is necessary to choose how many and which of these boundary pipes must be closed with N_{bv} gate valves or, equally, must be used for installing $N_{fm} = (N_{ec} - N_{bv})$ flow meters. In other terms, once found the possible positions e_{ij} (boundary pipes between clusters) for flow meters and boundary valves by spectral clustering (phase 1) and chosen the number of flow meters N_{fm} to be inserted in the network, the pipes, that must be closed, must be previously chosen among all the possible combinations of WNP layouts N_L expressed by binomial coefficient:

$$N_L = \binom{N_{ec}}{N_{fm}} \quad (1)$$

N_L can become, already with ordinary dimension of WNS, a huge number also for a small number k of DMAs.

While the first phase (clustering) does not modify the water system because no devices (flow meters or gate valves) are required, the second phase (dividing) can significantly change the network layout reducing the topological and energy redundancy [11] and, consequently, worsening the hydraulic performance.

Therefore, an optimization technique has been developed, in order to find, once fixed the number of flow meters N_{fm} , the best solution in the choice of the pipes in which to insert gate valves, minimizing the alteration of hydraulic performance and the level of service for the users. This aim was achieved by a heuristic procedure carried out with a Genetic Algorithm (GA) developed by the authors [11], minimizing the following objective function:

$$\min \left(\gamma \sum_{i=1}^n (z_i + h_i) Q_i \right) \tag{2}$$

where γ is the specific weight of water, z_i , h_i and Q_i are, respectively, the geodetic elevation, the pressure and the water demand of the i -th node. The objective function corresponds to the total nodal power of the network [5].

The GA parameters were the following: each individual of the population is composed by a sequence of chromosomes corresponding to the number of pipes belonging to the set N_{ec} . Each chromosome assumes value 1 if a gate valve will be inserted in the j -th pipe otherwise value 0 if a flow meter will be inserted. GA was carried out with 100 generations and with a population consisting of 500 individuals with a crossover percentage equal to $P_{cross} = 0.8$.

3 Case study

The city of Parete, with 10,800 inhabitants, is located in a densely populated area southern of Caserta (Italy). The water network has two sources and its main topological and energy characteristics are reported in Table 1 and Table 2, respectively. The hydraulic performance was evaluated using the commercial software EPANET2 [25] that numerically solves the non-linear hydraulic equations of the water system.

Table 1: Topological characteristics of Parete water distribution network

m	n	q	k	APL	D	λ_2	$\Delta\lambda$
[–]	[–]	[–]	[–]	[–]	[–]	[–]	[–]
282	184	0.017	3.05	8.80	20	0.021	0.062

The network has $m = 282$ links and $n = 184$ nodes and, from a topological point of view, in agreement with most large-scale real networks nature, it is a sparse network, so it is not fully connected and its number of edges $m \ll n^2$, with a link density value $q = 0.017$. Since the number of edges that can be connected to a single node is limited by the physical space in spatial networks [3], average node degree $k = 3.05$ is small.

The case study shows a small average path length $APL = 8.80$, presenting itself as a cohesive and robust network as well as the value of graph diameter $D = 20$ shows that the nodes are mutually and easily reachable and that the network are ordered in a decentralized fashion [29], which is an important aspect for an efficient communication (information flow) in a network. Concerning the main spectral measurements, the spectral gap $\Delta\gamma$ is equal to 0.062 and the algebraic connectivity λ_2

Table 2: Energy characteristics of Parete water distribution network.

h^*	h_{min}	h_{mean}	h_{max}	I_r
[m]	[m]	[m]	[m]	[—]
25.00	21.36	31.05	50.47	0.351

is equal to 0.021, so they assume low values, showing that the graph arrangement can be decomposed into isolated parts (clusters or districts) [13].

The hydraulic performance of Parete network, reported in the Table 2, is good in terms of maximum and mean nodal pressure, with h_{max} and h_{mean} higher than the design pressure $h^* = 25m$ (the pressure required to satisfy water demand at all nodes), but is not good with reference to minimum pressure h_{min} , that is significantly lower. Consequently, a low value of resilience index I_r [27], a global performance index measuring the surplus of energy compared to the energy strictly needed to satisfy nodal demand, results, indicating a low availability of the water system to be partitioned or, in other terms, to change its original layout by the insertion of gate valves without a decrease in hydraulic performance [17].

The first phase of proposed methodology generates a spectral clustering in $k = 2$ DMA (or cluster), as highlighted in the Figure 1, in which the bipartition of the network nodes of the graph according to the positive and negative components x_{v2} of the eigenvector v_2 is illustrated, with reference to diameter-weight with the $Ncut$ criterion.

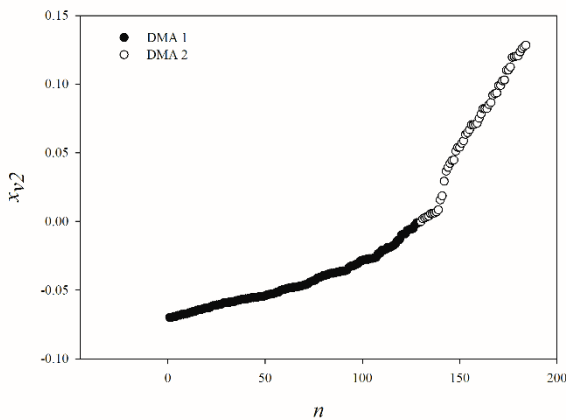


Fig. 1: Second smallest eigenvector v_2 , in increasing order vs nodes, separated in negative and positive components x_{v2} by referring to the diameter-weight with the $Ncut$ criterion.

This solution is in compliance with the continuity of both clusters DMA1 and DMA2. The bipartition of network nodes into two clusters is evident in Figure 1; indeed, the positive/negative eigenvector components x_{v2} are aligned in two different ways.

The results in terms of topological metrics are reported, for each weight combinations, in the Table 3 with reference to $Acut$ and $Ncut$ criteria, indicating: the number of districts k , the number of node n_k for each i -DMA, the number of edge cut N_{ec} , the number of flow meters N_{fm} and the number of gate valves $N_{bv} = N_{ec} - N_{fm}$.

In all cases, the two clusters are not perfectly balanced as number of nodes, highlighting an inherent inhomogeneity of the network.

Table 3: Partitioning indices for $k = 2$ DMAs with *Acut* and *Ncut* criteria.

Clustering method	Weight	k	n_{DMA1}	n_{DMA2}	N_{ec}	N_{bw}	N_{fm}
	[−]	[−]	[−]	[−]	[−]	[−]	[−]
<i>Acut</i>	L_{NW}	2	52	132	9	7	2
	L_D	2	69	115	10	8	2
	L_L	2	60	124	13	11	2
<i>Ncut</i>	LN_{NW}	2	53	131	10	8	2
	LN_D	2	55	129	9	7	2
	LN_L	2	60	124	12	10	2

Concerning the number of edge-cut N_{ec} , the best solutions correspond to the no-weight and diameter-weight both for L_{NW} ($N_{ec} = 9$ with *Acut*) and LN_{NW} ($N_{ec} = 10$ with *Ncut*) and for L_D ($N_{ec} = 10$ with *Acut*) and LN_D ($N_{ec} = 9$ with *Ncut*); while the worst ones correspond to the length-weight L_L ($N_{ec} = 13$ with *Acut*) and LN_L ($N_{ec} = 12$ with *Ncut*). Clearly, a low value of N_{ec} obtained at the end of the clustering phase can help in the second phase, aimed to find the positions of boundary valves and flow meters. Indeed a lower number of N_{ec} eases a small alteration of hydraulic performance of the network, because a smaller number of gate valves is then required. Further, a lower value of N_{ec} allows also an economic saving with fewer devices to be installed.

In Table 3, the number of flow meters is fixed for all combinations ($N_{fm} = 2$); it is the minimum possible number which guarantees the hydraulic performance of the network, at the same time simplifying the identification of water losses [20]. Clearly, the number of gate valves is equal to the difference $N_{bv} = N_{ec} - N_{fm}$.

After weighted spectral clustering, dividing phase was achieved, computing all the hydraulic performance metrics reported in Table 4.

The best solution corresponds to diameter-weight; in particular to LN_D obtained with *Ncut* criterion, with a deviation resilience index I_{rd} [6], that measures the reduction of the resilience index after WNP, equal to $I_{rd} = 5.13\%$, even with a slight increase of nodal minimum pressure $h_{min} = 22.44m$ compared to the network before partitioning (the increase of minimum pressure h_{min} is balanced from a very slight reduction of $h_{mean} = 31.00m$ and $h_{max} = 50.16$).

The worst solution corresponds to length-weight with *Ncut* criterion LN_L , which shows the maximum loss of hydraulic performance, with a resilience deviation index $I_{rd} = 54.13\%$.

In general, worse solutions correspond to the layouts with a higher value of N_{ec} ; while in the case of diameter-weight both *Acut* and *Ncut* found $N_{ec} = 9$, although the second one was significantly better in the second phase of network dividing. Anyway, although in the clustering phase with LN_{NW} and LN_D the number was the same ($N_{ec} = 9$), in the second phase, water network dividing was significantly better with LN_D , because the weight combination was crucial to define different clusters with a different number of nodes and different boundary edges, that allow to find a better solution with GA.

For the presented case study, it is clear that, from both topological and hydraulic point of view, the best solutions come from the diameter-weight combinations, with a smaller deterioration of hydraulic performance corresponding to *Ncut* criterion, which in fact leads to a balanced clustering taking into account the weights of the pipes [26].

Table 4: Energy Indices for $k = 2$ DMAs with $Acut$ and $Ncut$ criteria

Clustering method	Weight	I_r	I_{rd}	h_{min}	h_{max}	h_{mean}
	$[-]$	$[-]$	$[%]$	$[m]$	$[m]$	$[m]$
$Acut$	L_{NW}	0.292	16.81	17.48	50.28	29.78
	L_D	0.199	43.30	18.32	51.21	27.51
	L_L	0.217	38.18	17.46	50.28	28.04
	LN_{NW}	0.190	45.87	17.55	51.12	27.37
$Ncut$	LN_D	0.333	5.13	22.44	50.16	31.00
	LN_L	0.161	54.13	17.62	50.34	26.85

The case study was further extended to the analysis of a water network partitioning with $k = 4$ districts, by using the recursive spectral clustering [26] starting from the previously obtained two DMAs. In other words, each cluster DMA1 and DMA2 was again clustered into DMA11 and DMA12 and DMA21 and DMA22, respectively.

In Figure 2, the bipartition with the positive and negative components of the second smallest eigenvector of the network nodes of each previous cluster are illustrated, with reference to the diameter-weight with the $Ncut$ criterion. Also this solution is clearly in compliance with the continuity of nodes of each cluster.

In the case of the second bipartition, the Figure 2 does not show the same behavior highlighted in the Figure 1. In the case of DMA1 the difference between values of eigenvector components x_{v2} is evident while in the second case of DMA2 is very small. Further, in both cases of Figure 2, the eigenvalue components do not show an evident slope variation as in the Figure 1.

Concerning the number of edge-cuts N_{ec} , the best solution corresponds, also in this case with $k = 4$, to the diameter-weight L_D ($N_{ec} = 18$), while the worst corresponds to the length L_L and diameter LN_D weight combinations (in both cases $N_{ec} = 23$), as reported in the Table 5.

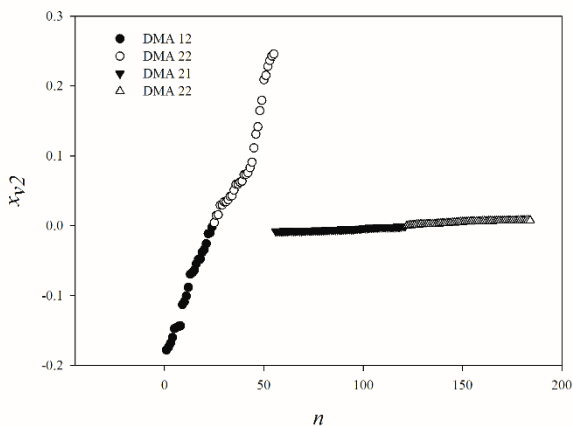


Fig. 2: Second smallest eigenvectors v_2 , in increasing order vs nodes, separated in negative and positive components x_{v2} by referring to the diameter-weight with the $Ncut$ criterion for DMA1 and DMA2.

Table 5: Partitioning indices for $k = 2$ DMAs with *Acut* and *Ncut* criteria.

Clustering method	Weight k		n_{DMA11}	n_{DMA12}	n_{DMA12}	n_{DMA12}	N_{ec}	N_{bw}	N_{fm}
	[-]	[-]	[-]	[-]	[-]	[-]	[-]	[-]	[-]
<i>Acut</i>	L_{NW}	4	27	25	63	69	21	15	6
	L_D	4	24	45	53	62	18	12	6
	L_L	4	25	35	52	72	23	17	6
	LN_{NW}	4	23	30	62	69	21	15	6
<i>Ncut</i>	LN_D	4	25	35	52	72	23	17	6
	LN_L	4	22	38	61	63	21	15	6

In Table 5, the number of flow meters is fixed for all combinations ($N_{fm} = 6$) in compliance with the hydraulic performance of the network.

For the dividing phase, all hydraulic performance metrics are reported in Table 6.

The best solution corresponds, also in this case, to diameter-weight LN_D with *Ncut*, with a deviation resilience index equal to $I_{rd} = 2.28\%$, with a very slight decrease of nodal minimum pressure $h_{min} = 21.19m$, nodal maximum pressure $h_{max} = 50.53m$ and nodal mean pressure $h_{mean} = 30.69m$ compared with the original network before partitioning. The result for 4 DMAs, with $I_{rd} = 2.28\%$, is better than previous result for 2 DMAs ($I_{rd} = 5.13\%$) because the number of flow meters (corresponding to opened boundary pipes) is higher in the second case ($N_{fm} = 6$ vs $N_{fm} = 2$).

The worst solution corresponds again to length-weight combination LN_L , which shows the maximum performance deterioration, with a resilience deviation index $I_{rd} = 30.20\%$, with also the lower value of the minimum pressure $h_{min} = 17.90m$. Generally, in terms of nodal pressure, the results are however good, as h_{mean} in each case is slightly lower than the original network value. Also in this second phase, for $k = 4$ DMAs, it is clear that, from both topological and hydraulic point of view, the best solution corresponds to the diameter-weight combinations with *Ncut* criterion.

Table 6: Energy Indices for $k = 2$ DMAs with *Acut* and *Ncut* criteria

Clustering method	Weight	I_r	I_{rd}	h_{min}	h_{max}	h_{mean}
	[-]	[-]	[%]	[m]	[m]	[m]
<i>Acut</i>	L_{NW}	0.306	12.82	18.51	50.26	30.03
	L_D	0.293	16.52	20.18	50.49	29.36
	L_L	0.300	14.53	20.23	50.38	29.79
	LN_{NW}	0.307	12.54	19.48	50.27	30.02
<i>Ncut</i>	LN_D	0.343	2.28	21.19	50.53	30.69
	LN_L	0.245	30.20	17.90	50.43	28.41

Finally, Figure 3 shows the Parete WNP in the case of two and four DMAs, corresponding to the best solutions LN_D in terms of minimum alteration of hydraulic performance (i.e. resilience deviation index).

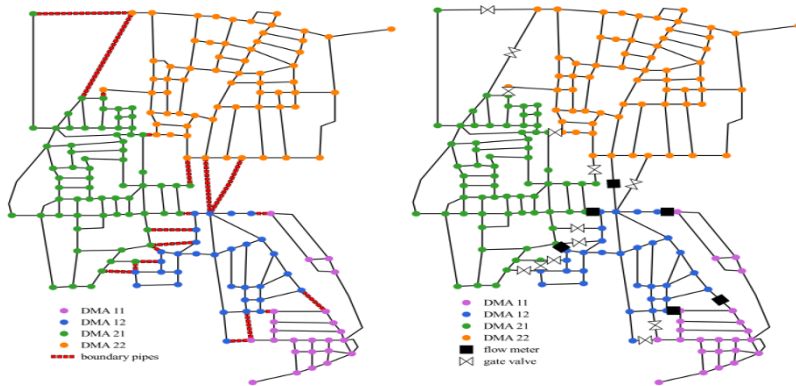


Fig. 3: Parete WSN partitioning in 4 DMA: clustering phase (left side) and dividing phase (right side).

In the left side of Figure 3, the first clustering phase is reported highlighting the edge-cuts (dashed lines); while, in the right side, the second dividing phase is illustrated, highlighting the positioning of optimal devices which ensures the minimum hydraulic performance deterioration.

4 Conclusion

The paper presents a preliminary application to a real water distribution network of weighted spectral clustering methods for water network partitioning, that represents one of the most innovative techniques to improve water supply network management.

Simulation results, obtained with different methods (*Acut* and *Ncut*) and pipe weights (no-weight, diameter and length) and for a different number of DMAs ($k = 2$ and $k = 4$), confirm the effectiveness of the procedure, highlighting that the best solution was diameter-weighted combination with *Ncut* method and a recursive spectral clustering, obtained according to the sign of the components of the eigenvector corresponding to the second smallest eigenvalue.

Further studies are in progress to improve the obtained results, integrating spectral methods with other clustering and graph partitioning algorithms and testing the procedure on larger water networks, also adopting other hydraulic and geometric information as weight combinations.

References

- [1] Alonso, J.M., Alvarruiz, F., Guerrero, D., Hernández, V., Ruiz, P.A., Vidal, A.M., Martínez, F., Vercher, J., Ulanicki, B.: Parallel computing in water network analysis and leakage minimization. *Journal of Water Resources Planning and Management* **126**(4), 251–260 (2000)
- [2] Barthélemy, M., Flammini, A.: Modeling urban street patterns. *Physical review letters* **100**(13), 138,702 (2008)
- [3] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: Structure and dynamics. *Physics reports* **424**(4), 175–308 (2006)
- [4] Carvalho, R., Buzna, L., Bono, F., Gutiérrez, E., Just, W., Arrowsmith, D.: Robustness of trans-european gas networks. *Physical review E* **80**(1), 016,106 (2009)

- [5] Di Nardo, A., Di Natale, M.: A heuristic design support methodology based on graph theory for district metering of water supply networks. *Engineering Optimization* **43**(2), 193–211 (2011)
- [6] Di Nardo, A., Di Natale, M.: A heuristic design support methodology based on graph theory for district metering of water supply networks. *Engineering Optimization* **43**(2), 193–211 (2011)
- [7] Di Nardo, A., Di Natale, M., Giudicianni, C., Musmarra, D., Santonastaso, G.F., Simone, A.: Water distribution system clustering and partitioning based on social network algorithms. *Procedia Engineering* **119**, 196–205 (2015)
- [8] Di Nardo, A., Di Natale, M., Greco, R., Santonastaso, G.: Ant algorithm for smart water network partitioning. *Procedia Engineering* **70**, 525–534 (2014)
- [9] Di Nardo, A., Di Natale, M., Musmarra, D., Santonastaso, G.F., Tzatchkov, V., Alcocer-Yamanaka, V.H.: Dual-use value of network partitioning for water system management and protection from malicious contamination. *Journal of Hydroinformatics* **17**(3), 361–376 (2015)
- [10] Di Nardo, A., Di Natale, M., Santonastaso, G., Tzatchkov, V., Alcocer-Yamanaka, V.: Water network sectorization based on a genetic algorithm and minimum dissipated power paths. *Water Science and Technology: Water Supply* **13**(4), 951–957 (2013)
- [11] Di Nardo, A., Di Natale, M., Santonastaso, G.F., Venticinque, S.: An automated tool for smart water network partitioning. *Water resources management* **27**(13), 4493–4508 (2013)
- [12] Diao, K., Zhou, Y., Rauch, W.: Automated creation of district metered area boundaries in water distribution systems. *Journal of Water Resources Planning and Management* **139**(2), 184–190 (2012)
- [13] Estrada, E.: Network robustness to targeted attacks. the interplay of expansibility and degree distribution. *The European Physical Journal B-Condensed Matter and Complex Systems* **52**(4), 563–574 (2006)
- [14] Ferrari, G., Savic, D., Becciu, G.: Graph-theoretic approach and sound engineering principles for design of district metered areas. *Journal of Water Resources Planning and Management* **140**(12), 04014.036 (2013)
- [15] Fiedler, M.: Algebraic connectivity of graphs. *Czechoslovak mathematical journal* **23**(2), 298–305 (1973)
- [16] Gomes, R., Sá Marques, A., Sousa, J.: Identification of the optimal entry points at district metered areas and implementation of pressure management. *Urban Water Journal* **9**(6), 365–384 (2012)
- [17] Greco, R., Di Nardo, A., Santonastaso, G.: Resilience and entropy as indices of robustness of water distribution networks. *Journal of Hydroinformatics* **14**(3), 761–771 (2012)
- [18] Herrera, M., Canu, S., Karatzoglou, A., Pérez-García, R., Izquierdo, J.: An approach to water supply clusters by semi-supervised learning. *Proceedings of International Environmental Modelling and Software Society (IEMSS)* (2010)
- [19] Izquierdo, J., Herrera, M., Montalvo, I., Pérez-García, R.: Division of Water Supply Systems into District Metered Areas Using a Multi-agent Based Approach, pp. 167–180. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)
- [20] Limited, U.W.I.R.: A Manual of DMA Practice. UK Water Industry Research Limited (1999)
- [21] Mays, L.W.: Water distribution system handbook. McGraw-Hill Professional Publishing New York, NY, USA (1999)
- [22] Nardo, A.D., Natale, M.D., Santonastaso, G., Tzatchkov, V., Yamanaka, V.A.: Divide and conquer partitioning techniques for smart water networks. *Procedia Engineering* **89**, 1176 – 1183 (2014). DOI <http://dx.doi.org/10.1016/j.proeng.2014.11.247>. URL <http://www.sciencedirect.com/science/article/pii/S1877705814023625>
- [23] Newman, M.E.: The structure and function of complex networks. *SIAM review* **45**(2), 167–256 (2003)
- [24] Perelman, L.S., Allen, M., Preis, A., Iqbal, M., Whittle, A.J.: Automated sub-zoning of water distribution systems. *Environmental Modelling & Software* **65**, 1–14 (2015)
- [25] Rossman, L., for Environmental Research Information (Estats Units d'Àmerica), C., d'Àmerica. Environmental Protection Agency. Office of Research, E.U., Development,

- d' Amèrica), N.R.M.R.L.E.U.: Epanet 2: users manual. U.S. Environmental Protection Agency. Office of Research and Development. National Risk Management Research Laboratory (2000). URL <https://books.google.it/books?id=CgijMwEACAAJ>
- [26] Shi, J., Malik, J.: Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* **22**(8), 888–905 (2000)
- [27] Todini, E.: Looped water distribution networks design using a resilience index based heuristic approach. *Urban water* **2**(2), 115–122 (2000)
- [28] Tzatchkov, V.G., Alcocer-Yamanaka, V.H., Ortíz, V.B.: Graph theory based algorithms for water distribution network sectorization projects. In: Proc. of the 8th Annual Water Distribution Systems Analysis Symposium WDSA, Cincinnati, Ohio, USA (2006)
- [29] Yazdani, A., Jeffrey, P.: Robustness and vulnerability analysis of water distribution networks using graph theoretic and complex network principles. *Proceeding of Water Distribution System Analysis 2010* pp. 12–15 (2010)
- [30] Yazdani, A., Jeffrey, P.: Complex network analysis of water distribution systems. *Chaos* **21**(1) (2011)

Robust optimization of power network operation: storage devices and the role of forecast errors in renewable energies

Carsten Matke, Daniel Bienstock, Gonzalo Muñoz, Shuoguang Yang, David Kleinhans and Sebastian Sager

Abstract In this paper we investigate a robust optimization framework for controlling energy storage devices in power networks with high share of fluctuating renewable energy sources. Our approach relies on the industry-standard DC power flow approximation, together with a multi-stage model that incorporates renewable uncertainty and an approximation of battery dynamics. More precisely, we consider storage device operation under linear control and taking into account power limits, energy conversion efficiencies, and energy limits for the state of charge. The aim of the robust optimization is to minimize costs for generating energy from conventional power generators while relying on storage to compensate for renewable output forecast errors. In order to obtain a solution we propose a cutting-plane procedure which can be used for investigating practical case studies.

Key words: robust optimization, power network control, fluctuating renewables, energy storage devices

1 Introduction

The ongoing transition from conventional to renewable energy sources (RES) is causing a dramatic impact on power generation. Since the late 19th century, power networks have been designed as centralized networks, where large power plants rely on high-voltage power transmission lines so as to transport energy from one region to another one [17]. Distributed substations transform voltage to lower levels suitable for electricity distribution to consumers. The transition to RES implies the (at least partial) substitution of large conventional power plants (e.g., coal and nuclear power plants) by many but smaller generation units (e.g., wind and solar farms). These RES generation

Carsten Matke (e-mail: carsten.matke@next-energy.de) · David Kleinhans
NEXT ENERGY • EWE Research Centre for Energy Technology at the University of Oldenburg,
Carl-von-Ossietzky-Str. 15, 26129 Oldenburg, Germany

Daniel Bienstock · Gonzalo Muñoz (e-mail: dano@columbia.edu) · Shuoguang Yang
Department of Industrial Engineering & Operations Research, Columbia University in the City of
New York, 500 West 120th Street, New York, NY 10027, USA

Sebastian Sager (e-mail: sager@ovgu.de)
Institute for Mathematical Optimization, Otto-von-Guericke-University Magdeburg, Univer-
sitätsplatz 2, 39106 Magdeburg, Germany

facilities are located at distributed sites according to the availability of resources and, hence, might be in regions with weak grid infrastructures and far away from electricity consumers. In addition, production from RES fluctuates in time. As a consequence, RES imply increased challenges for the grid infrastructure and its design, operation, and control.

In principle, both peak loads of the grid and the temporal fluctuations of the availability of resources can be reduced using storage devices [1, 21]. In this context, storage devices can reduce the need for investments in grid infrastructure and / or alternative generation capacities. The amount, location, and technology of storage devices required and their operation form a complex optimization problem and several contributions deal with different aspects of related problems [13, 14, 16, 18, 26]. See e.g., [11, 27], for discussions on wind power forecast errors, and [24] for a long-term model involving scenarios. Here, we aim to develop a model to study optimal control strategies for storage devices used to partly compensate for deviations from predicted renewable power production. For this purpose, we build on the well-known methodology of “robust optimization” [2, 4]. **We focus on the efficient solution of a robust model that accommodates nonconvexities in the modeling of battery operation while yielding a convex approach (Section (3.7)). The work isolates the interplay of uncertainty with battery operation, while not modeling several realistic grid operation details.**

In our contribution, we consider power networks with high share of RES, but still having conventional power generators both used as a backbone and so as to provide real-time and secondary frequency control. The objective of the optimization is minimizing generation costs of conventional power generators (the standard OPF, or Optimal Power Flow setup) while operating the network such that loads (i.e. demands) are all met and with high likelihood no line overloads occur and that battery operation remains safe.

The rest of the paper is organized as follows. Section 2 gives a practical problem which motivates our robust optimization framework while section 3 provides a detailed mathematical formulation and we propose a cutting-plane procedure on how to solve the robust optimization problem. In section 4 we show preliminary computational results and section 5 gives a conclusion and an outlook of further investigations.

2 Motivation

Before describing the formulation of our robust optimization framework, first let us consider the following simple example. The network shown in Figures 1 and 2 have the exact same network components in those seven buses. The quantities shown are in units of power (e.g. MW). This specific network contains one generator G that can produce 0 – 200 units, two buses L with 100 units of power load, one storage device B able to store 0 – 100 units of energy which starts at time zero with zero storage, and three renewable energy sources RES that have fluctuating generation in each time period.

Consider two periods. In the first period (Figure 1) we assume there is no uncertainty in the renewable power generation and in the second period (Figure 2) we assume the renewables produce between 0 and 20 units each, which expresses the uncertainty. A solution our framework would provide is the following:

In the first period, the power generator G outputs 170 units from which 100 units flow into the first bus L with power load 100 and 70 units are transmitted to the second bus L with power load 100. This power load bus obtains further 30 units from the renewables. Another 30 units from the renewables are used to charge the storage device B .

In the second period, the power generator G outputs 170 units. Note that using this generation levels, regardless of the renewable generation level, we can always respond to the power load using the storage device B which has stored 30 units from the first period.

Furthermore, the location of the storage device as shown in this illustrative example is the only logical one. The reader can check that all other locations of the storage device would result in an infeasible problem.

3 Robust optimization framework

A power transmission network is a graph where some nodes are sources of power flow (e.g., power generator, renewable) and some nodes are sinks representing loads. In power engineering practice nodes can hold both power sources and loads. Additionally, storage devices can be located at any node and can act as a power source while charging or discharging. In the power community nodes of the graph are called buses and the edges of the graph are called branches; these are circuits transmitting electrical power between buses. For a thorough introduction to power networks we refer the reader to textbooks (e.g., [6, 15, 25]), but a brief review of the heavily used “DC-OPF” to power flows is appropriate. Each bus k has a state variable θ_k that represents the voltage phase angle. In particular, given a branch between buses k and m , we have:

$$P_{km} = y_{km} (\theta_k - \theta_m), \tag{1}$$

where the parameter y_{km} is the *susceptance* of branch km . For thermal protection reasons, the absolute value of the flow on a branch is limited by a parameter known as the “limit”, or “rating” of the branch:

$$|y_{km} (\theta_k - \theta_m)| \leq L_{km}. \tag{2}$$

A final set of equations enforce *flow balance* at each bus (Kirchhoff’s law): the net outflow (flow leaving minus flow entering a bus) must equal to total generation minus total demand at that bus [23]:

$$\sum_j P_{kj} - \sum_j P_{jk} = P_k^g - P_k^d. \tag{3}$$

This review can be made more accurate so as to account for other electrical devices and phenomena, such as transformers and line charging (shunts). Equation (4) (see [28, p. 27 in eq. (3.32)]) given below summarizes the flow balance constraints. In this equation the matrix B is obtained by substituting (1) into (3) at each bus k :

$$B \theta^t = P_t^g - P_t^d. \tag{4}$$

Fig. 1 Period 1: No uncertainty in the renewable power forecast. In the optimal solution G generates 170 units and 30 units from renewables are charged into the storage device B . Flow capacities on each branch are indicated by purple colors.

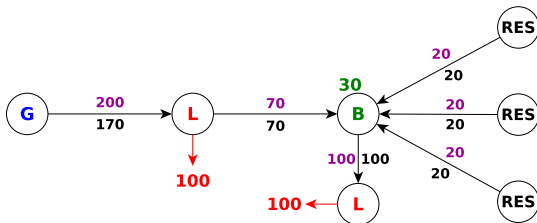
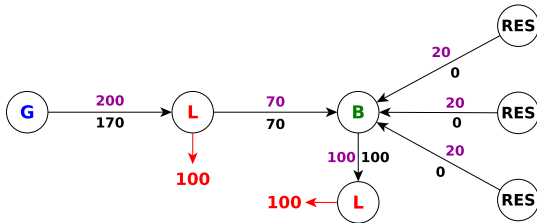


Fig. 2 Period 2: Incorporates uncertainty in renewable power forecast. In the optimal solution G generates 170 units and 30 units are discharged from the storage device B . Flow capacities on each branch are indicated by purple colors.



In this equation t specifies a time index, and P_t^g and P_t^d indicate, respectively, vectors of generation and loads at time t . For brevity we omit background on the (approximate) validity of (4). One obtains the following standard optimization problem, referred to as DC-OPF:

$$\min \sum_{k,t} c_{k,t} (P_{k,t}^g), \quad (5)$$

where the functions $c_{k,t}$ are convex quadratics cost function of power generation.

The solution of the problem with objective (5) and constraints (4), (2) (plus additional constraints on e.g. generation that are omitted here) provides phase angle variables θ , and through equations (1) yields the power flows, which thus depend on both the power generated and the power consumed at all buses and all times. A problem of this type is normally solved with some frequency, e.g. every five minutes, using *estimates* of the loads in the next time window. Real-time deviations of the loads from these forecasts, which are usually small, are handled through the mechanisms of primary and secondary frequency control.

3.1 Basic control model

Our control scheme is given by the following modification of (4), where we ignore shunts and transformers and leave out the time index t for simplicity:

$$B\theta = P^g - P^d - \left(\sum_i w_i \right) \lambda + \bar{w} + w, \quad (6)$$

In this equation, the term $\bar{w} + w$ corresponds to the power injected by the renewables; \bar{w} is the vector of forecast renewable power generation and w is the vector of deviations from the forecast. The term $(\sum_i w_i) \lambda$ corresponds to the power obtained from storage devices when using a linear control responsive to renewable power deviations. Here, and in what follows, bold face is used to indicate uncertain quantities (this includes θ , since in (6) the state variables of the voltage phase angles θ are dependent of w). Note that for buses k without RES \bar{w}_k and w_k are zero. In (6) λ is a vector of **decision variables**, with entries λ_i for each bus i (both fixed at zero for buses without storage device). Thus, with $\lambda \geq 0$, (6) indicates that storage devices absorb renewable power excesses (over the forecast) and balance out shortages.

- 1 In application of the control, the quantity $\sum_i w_i$ would be **estimated** from measurements at the start of the given time period and that value would be used throughout the period. Thus, (6) *would not* accommodate all real-time additional errors.
- 2 As stated the problem is (deliberately) unrealistic. For example, one should also account for errors in estimating loads, and ramping constraints on generators, the use of generators to counteract renewable variation, and the interaction of generators and storage. These omissions are purposely made so as to isolate the interaction of uncertainty and storage operation. However, note that implicitly our robust optimization problem will allow e.g. using generators to charge storage.

In a robust setting, we will choose an uncertainty model – this model will specify which forecast deviations w our approach will compensate for. When choosing a tuple (P^g, λ) we must ensure that (6) is a feasible system of equations for any allowable realization of w . In addition, it is explicitly required that (6) remains feasible in the nominal case, ($w = 0$), i.e. the system of equations:

$$B\theta^{\text{nom}} = P^g - P^d + \bar{w}, \quad (7)$$

is feasible. Since the sum of rows of B is zero, in order for (6) to be feasible, the sum of right-hand side values has to be zero, and likewise for (7). Applying this principle to (7) we obtain:

$$\sum_i (P_i^g - P_i^d + \bar{w}_i) = 0. \quad (8)$$

Using (8) and applying the principle to (6), we obtain:

$$\sum_i \lambda_i = 1, \quad (9)$$

which will be imposed as a constraint in the robust optimization problem. In addition, either (8) is explicitly included as a constraint or (7) is included as a subsystem.

However, ensuring that (6) is feasible for all allowable realizations of the deviations \mathbf{w} , is certainly not enough to guarantee stability. Two issues are left outstanding: guaranteeing that storage operation (across multiple time units) remains feasible, and guaranteeing that the thermal constraints (2) also hold. We will return to these issues later. At this point it is worth pausing to discuss the uncertainty model and its implications.

3.2 Storage device model

A storage device is a component in a power network which can be operated as energy source or energy sink. For the conversion from electrical energy to a storage device specific energy form (e.g., chemical energy) at bus k and time t , there are technology specific charging and discharging efficiencies $0 < \eta_{k,t}^c \leq 1$ and $0 < \eta_{k,t}^d \leq 1$, respectively. As a consequence, a *discharging* storage device with electrical power $P_{k,t}^B > 0$ discharges with a (chemical) power of $P_{k,t}^B / \eta_{k,t}^d$ inside the storage device. On the opposite, a *charging* storage device with electrical power $P_{k,t}^B < 0$ charges with a (chemical) power of $-\eta_{k,t}^c P_{k,t}^B$. Further, the electrical charging and discharging power are limited by $P_{k,t}^{B,\min} \leq P_{k,t}^B \leq P_{k,t}^{B,\max}$. The modeling of charging and discharging efficiencies introduces nonconvexities; in fact previous authors have resorted to the use of binary variables so as to accommodate this feature. However we will show that our proposed cutting-plane algorithm bypasses this complexity.

3.3 Uncertainty model

We denote by \mathcal{W} the set of forecast errors which need to be considered, $\mathbf{w} = \{\mathbf{w}_{k,t}\}$. In particular, we will consider *concentration models*, where there are matrices C^1, C^2 , both with non-negative entries, a vector b , and values $l_{k,t} \leq 0 \leq u_{k,t} \forall t \forall k$, such that \mathcal{W} is the set of vectors satisfying:

$$C^1 w^+ + C^2 w^- \leq b, \quad (10a)$$

$$l_{k,t} \leq \mathbf{w}_{k,t} \leq u_{k,t} \quad \text{for all } t \text{ and } k, \quad (10b)$$

where w^+ and w^- are, respectively, the vectors of component-wise values $\max\{w_j, 0\}$ and $\max\{-w_j, 0\}$.

An example, with $T = 1$ is where for values $\gamma_k \geq 0$ ($k = 1, \dots, N$) and $\Gamma \geq 0$, \mathcal{W} is the set of vectors satisfying

$$|\mathbf{w}_{k,1}| \leq \gamma_k \quad \forall k, \quad \sum_k \frac{|\mathbf{w}_{k,1}|}{\gamma_k} \leq \Gamma.$$

Note that model (10) allows for constraints across time periods, and for non-symmetries (e.g. $l_{k,t} \neq -u_{k,t}$ for some t and k).

In practice, good estimates for \mathcal{W} can either be obtained from literature [12] or from direct analysis of data on forecast and realized productions, which are available for several control zones.¹

3.4 Optimization model

We can now outline our optimization model. The decision variables are the quantities $P_{k,t}^g$ (for each generator at bus k and time period t) and $\lambda_{i,t}$ (for each storage device at bus i and time period t). It may be stated as

$$\min_{P^g, \lambda} \sum_t \sum_k c_{k,t}(P_{k,t}^g) \quad (11a)$$

s.t. the following constraints being feasible at all times t , for all $w \in \mathcal{W}$:

$$B \theta^t = P_t^g + \bar{w}_t + w_t - \left(\sum_i w_{i,t} \right) \lambda_t - P_t^d \quad (11b)$$

$$y_{km} |\theta_k^t - \theta_m^t| \leq L_{km} \quad \text{for all } km \quad (\text{line limits at time } t) \quad (11c)$$

$$\text{battery operation constraints} \quad (11d)$$

We stress that the only decision variables in this problem are the P^g and λ . In sections below we will deal with constraint (11d). Note that problem (11) is an extension of the standard DC-OPF so as to incorporate the linear storage control model and the uncertainty in renewable output. As we indicated above, the solution to problem (11) would *not* provide 100% protection against renewable output uncertainty; however through judicious construction of the uncertainty model \mathcal{W} it could be used to account for *most* of the uncertainty, with smaller, leftover errors left to be handled by standard primary and secondary control. See e.g. [7].

3.5 Algorithm

Problem (11) is a convex-objective problem but possibly including nonconvexities due to the uncertainty model and the battery operation model. However, we will show that the problem can in fact be very efficiently solved as a sequence of *linearly constrained* problems, thereby avoiding nonconvexities. The algorithm relies on the extremely effective paradigm of *cutting-plane* or Benders' decomposition methods [3] (see [5, 7, 8, 9, 10]). In the context of power engineering, works closely related to this paper are [19] and [20], discussed above.

The algorithm iterates by maintaining a linear relaxation of the constraints (11b)-(11d), that we shall term the *working formulation*. At the start of the algorithm the working formulation consists of constraints (8) and (9). Then, this formulation is iteratively enriched. Let us denote the working formulation in iteration K as $A^K P^g + B^K \lambda \geq b^K$ for appropriate matrices A^K , B^K and vector b^K . At iteration K the algorithm proceeds as follows:

Step 1. Solve $\min_{P^g, \lambda} \sum_t \sum_k c_{k,t}(P_{k,t}^g)$, subject to $A^K P^g + B^K \lambda \geq b^K$.

Let P_b^* , λ^* be an optimal solution (of this relaxation).

Step 2. **Either** show that P_b^* , λ^* satisfies constraints (11b) - (11d) for all $w \in \mathcal{W}$,
or find an inequality

¹ <http://www.tennetso.de/site/de/Transparenz/veroeffentlichungen/netzkennzahlen/tatsaechliche-und-prognostizierte-windenergieeinspeisung>

$$\alpha P^g + \beta \lambda \geq \alpha_0 \quad (12)$$

which is valid for problem (11) but violated by P_b^*, λ^* . In the latter case, add inequality (12) to the working formulation.

Step 2 is the so-called “separation procedure”. If the “either” case holds we have computed an optimal solution to problem (11). If the “or” holds the inequality we have added cuts-off the vector P_b^*, λ^* and so this vector will be excluded in all future iterations of the algorithm.

The separation procedure is thus at the heart of our algorithm. We have seen that imposing (8) and (9) for each t guarantees that (11b) will always be satisfied by a pair P_b^*, λ^* . Hence whenever the “or” case of Step 2 applies it must be the case that a constraint (11c) or (11d) is violated. In the next sections we will outline how the separation procedure is to be implemented in the case of these constraints. In both cases, the key step will be to compute, given a pair P_b^*, λ^* , a **worst-case** vector \mathbf{w} which is chosen so as to maximize the violation, by P_b^*, λ^* , of some constraint (11c) or (11d). This is the so-called **adversarial problem**. Having solved the adversarial problem, the task of constructing an appropriate violated cut (12) will also be shown to be straightforward. In Sections 3.6 and 3.7 we will indicate the specific form of the cuts that we use.

References [19] and [20] present related algorithms for problems involving storage. A primary difference between those works and ours concerns the modeling of nonconvex battery operation, which in particular we handle in an efficient manner as detailed in the next section.

3.6 Adversarial problem for storage device operation

As discussed above, for a storage device at bus k at time t we have:

$$\Delta_t P_{k,t}^B = \Delta_t P_{k,t}^B(\lambda, \mathbf{w}) = -\lambda_{k,t} \sum_j \mathbf{w}_{t,j},$$

where Δ_t is the length of period t . Then, the energy level at storage device k at time t , for a given \mathbf{w} , is:

$$E_{k,t}(\mathbf{w}) \doteq E_{k,0} + \sum_{i=1}^t \Delta_i \left(-(\eta_{k,i}^d)^{-1} [P_{k,i}^B(\lambda, \mathbf{w})]^+ + \eta_{k,i}^c [P_{k,i}^B(\lambda, \mathbf{w})]^- \right), \quad (13)$$

where for a real x , we denote $x^+ = \max\{x, 0\}$ and $x^- = \max\{-x, 0\}$. The logic for equation (13) is simple. Suppose first that $P_{k,i}^B > 0$. Then, the storage device is discharging at time i , moreover the second term in the sum in (13) is zero. Similarly, when $P_{k,i}^B < 0$. Continuing with (13), we have:

$$E_{k,t}(\mathbf{w}) = E_{k,0} + \sum_{i=1}^t \Delta_i \lambda_{k,i} \left(-(\eta_{k,i}^d)^{-1} \left[\sum_j \mathbf{w}_{j,i} \right]^- + \eta_{k,i}^c \left[\sum_j \mathbf{w}_{j,i} \right]^+ \right). \quad (14)$$

This is a **nonconvex** model. However, we can optimize over it efficiently. We need to ensure that for all storage buses k , for all t , and for all $\mathbf{w} \in \mathcal{W}$:

$$E_{k,t}^{\min} \leq E_{k,0} + \sum_{i=1}^t \Delta_i \lambda_{k,i} \left(-(\eta_{k,i}^d)^{-1} \left[\sum_j \mathbf{w}_{j,i} \right]^- + \eta_{k,i}^c \left[\sum_j \mathbf{w}_{j,i} \right]^+ \right) \leq E_{k,t}^{\max},$$

or, in other words, for all k and t ,

$$\max_{\mathbf{w} \in \mathcal{W}} \left\{ \sum_{i=1}^t \Delta_i \lambda_{k,i} \left(-(\eta_{k,i}^d)^{-1} \left[\sum_j \mathbf{w}_{j,i} \right]^- + \eta_{k,i}^c \left[\sum_j \mathbf{w}_{j,i} \right]^+ \right) \right\} \leq E_{k,t}^{\max} - E_{k,0}, \tag{15}$$

and

$$\min_{\mathbf{w} \in \mathcal{W}} \left\{ \sum_{i=1}^t \Delta_i \lambda_{k,i} \left(-(\eta_{k,i}^d)^{-1} \left[\sum_j \mathbf{w}_{j,i} \right]^- + \eta_{k,i}^c \left[\sum_j \mathbf{w}_{j,i} \right]^+ \right) \right\} \leq E_{k,t}^{\min} - E_{k,0}. \tag{16}$$

To help with this task we have the following Lemma:

Lemma 3.1. *Suppose $w \in \mathcal{W}$. Define the vectors \hat{w} and \check{w} as follows:*

$$\hat{w}_{j,i} = \begin{cases} 0, & \text{if } \sum_j w_{j,i} < 0, \\ w_{j,i}, & \text{otherwise} \end{cases}$$

$$\check{w}_{j,i} = \begin{cases} 0, & \text{if } \sum_j w_{j,i} > 0. \\ w_{j,i}, & \text{otherwise} \end{cases}$$

Then **(a)** $\check{w} \in \mathcal{W}$ and $\hat{w} \in \mathcal{W}$, and **(b)** for any k and t , $E_{k,t}(\check{w}) \leq E_{k,t}(w) \leq E_{k,t}(\hat{w})$.

Statement (a) follows from the assumptions about the disturbances \mathcal{W} , and (b) rests on expression (14).

Thus, consider expression (15). Using Lemma 3.1 this expression is equivalent to:

$$E_{k,t}^{\max} - E_{k,0} \geq \max \left\{ \sum_{i=1}^t \Delta_i \lambda_{k,i} \left(\eta_{k,i}^c \sum_j w_{j,i} \right) \right\} \tag{17}$$

s.t. $w \in \mathcal{W}$
 $\sum_j w_{j,i} \geq 0, \quad \text{for } i = 1, 2, \dots, t$

Given a choice of control parameters λ , this is an optimization problem that is readily solved for most robust models e.g., the concentration model (10). Expression (16) is similarly handled. As a consequence, suppose that λ^* is a particular control vector which fails to satisfy e.g (15). Using (17) we can compute $\hat{w} \in \mathcal{W}$, with $\hat{w} \geq 0$, and such that

$$\left\{ \sum_{i=1}^t \Delta_i \lambda_{k,i}^* \left(\eta_{k,i}^c \left[\sum_j \hat{w}_{j,i} \right]^+ \right) \right\} > E_{k,t}^{\max} - E_{k,0}.$$

But since \hat{w} is an allowable vector of deviations, we must have that any feasible λ , as just discussed, satisfies

$$\left\{ \sum_{i=1}^t \Delta_i \lambda_{k,i} \left(\eta_{k,i}^c \left[\sum_j \hat{w}_{j,i} \right]^+ \right) \right\} \leq E_{k,t}^{\max} - E_{k,0}. \tag{18}$$

In other words (18) is a valid inequality which the control λ^* satisfies. In summary, given λ the solution of problem (17) will decide whether λ is feasible or not, and in the latter case we will also obtain an inequality violated by λ .

3.7 Adversarial problem for line limit constraints

In the adversarial problem, we assume that we have chosen specific values P^{g*} and λ^* for the vectors P^g and λ , and we are trying to pick the vector w so that the flow in some line becomes too large (larger than the flow capacity) in absolute value. Consider a line km , where for simplicity we are assuming that $k \neq \rho$ and $m \neq \rho$. The enumerators goal is to pick w so to make $|\theta_k - \theta_m|$ large, i.e. larger than the L_{km}/y_{km} (here L_{km} is the flow capacity on branch km). Let us focus on $\theta_k - \theta_m$ for simplicity. Then the adversarial problem is as follows:

$$\begin{aligned} \text{ADV1: } \max \quad & \theta_k - \theta_m \\ \text{s.t. } \quad & B\theta = P^{g*} - P^d - \left(\sum_i w_i \right) \lambda^* + \bar{w} + w \\ & w \in \mathcal{W} \end{aligned} \quad (19)$$

In this optimization problem, as in (17), the only variables are the w . Also notice that we are using the power flow equations (under uncertainty) in the form (6).

We assume that our network is connected. Thus B has rank $n - 1$ where $n =$ number of buses, and as a system of equations on θ , (6) has one degree of freedom. Using standard methods we obtain an equivalent restatement of (19):

$$\theta = V \left(P^{g*} - P^d + \bar{w} \right) - \left(\sum_i w_i \right) V \lambda^* + V w,$$

where V is an appropriate matrix (a pseudo-inverse of B). Denote by v_j the j^{th} row of V , for any j . Then

$$\theta_k - \theta_m = (v_k - v_m) \left(P^{g*} - P^d + \bar{w} \right) - (v_k - v_m) \lambda^* \left(\sum_i w_i \right) + (v_k - v_m) w,$$

and so **ADV1** can be equivalently rewritten as:

$$\begin{aligned} \text{ADV2: } \max \quad & (v_k - v_m) \left(P^{g*} - P^d + \bar{w} \right) - (v_k - v_m) \lambda^* \left(\sum_i w_i \right) + (v_k - v_m) w \\ \text{s.t. } \quad & w \in \mathcal{W} \end{aligned}$$

Formulation **ADV2** can be used implicitly so as to generate cuts, as in Step 2 of the algorithm in Section 3.5, as follows. Suppose that the value of **ADV2** is too high, i.e. it is greater than L_{km}/y_{km} . Denote the optimal solution to **ADV2** by w^o ; this is a function of (P^{g*}, λ^*) . Then of course $w^o \in \mathcal{W}$ and by definition:

$$(v_k - v_m) \left(P^{g*} - P^d + \bar{w} \right) - (v_k - v_m) \lambda^* \left(\sum_i w_i^o \right) + (v_k - v_m) w^o > L_{km}/y_{km}. \quad (20)$$

From these observations we can derive a cut that separates (P^{g*}, λ^*) from the set of feasible solutions to the robust problem: suppose P^g, λ is an arbitrary feasible solution to the robust problem. It follows that, should the adversary use renewable power deviations w^o , the adversary will fail, i.e.:

$$(v_k - v_m) \left(P^g - P^d + \bar{w} \right) - (v_k - v_m) \lambda \left(\sum_i w_i^o \right) + (v_k - v_m) w^o \leq L_{km}/y_{km}. \quad (21)$$

In other words, (21) is an inequality that every feasible solution must satisfy. This inequality (violated by (P^{g*}, λ^*) as per (20)) is added as cut to the working formulation and completes the iteration.

4 Preliminary computational experiments

Here we outline ongoing experiments using the well-known “Polish grid 2003-2004 winter peak” dataset available through MATPOWER [29]. This realistic and meshed transmission system has 2746 buses, 3514 branches, 388 generators, and a total load of approximately 25GW. To construct instances of our problems on this data set, we placed wind farms and storage devices at the buses holding the 50 largest generators. The rationale for this is that such buses are likely to be attached to strong (i.e. large capacity) branches; in this way we avoid spurious infeasibility conditions. The wind farms on average imply a (large) 38% renewable penetration, whereas the batteries at their maximum energy level would account for 50% of all loads. (We note that such high battery capacities are needed in order to match large forecast errors). We thus used a parameter, “w-scale” to scale up or down all wind farms, and another parameter, “b-scale” to likewise scale batteries. The following table outlines results for the single time period case of our model.

Table 1: One-period results

w-scale	b-scale	Γ	Cost	Iterations	Time (s)
0.5	0.5	2	1238690.5	4	6.75
0.5	0.5	10	1242090.94	8	12.59
0.5	0.5	20	1243021.71	7	10.87
1.0	0.5	10	infeasible	1	1.97
1.0	0.1	32	infeasible	1	1.94

In Table 2 we outline results for a six-period problem using the same network, wind farm, and battery setup as above. Loads grow at an approximate rate of 1% per period.

Table 2: Six-period results

penetration	Γ	Cost	Iterations	Time (m)
12 %	5	8145315	24	5
12 %	8	8165998	75	12
17 %	10	infeasible	64	10

5 Conclusions

In our contribution we considered power networks with high shares of RES integrated with a reasonable amount conventional power generators, which can be operated in a flexible manner. For these systems, we motivated and developed a new general approach, which optimizes the commitment of power generators and storage devices taking into account both capacity limits in the grid and the generators and deviations of the renewable power production from its prediction. Using a methodology known as “robust optimization” we were able to suggest an optimization framework, which guarantees the robustness of the operation strategy for generators and storage devices with respect to forecast errors in the predicted renewable power generation. We showed that by means of the *cutting-plane* method [3] an effective algorithm solves instances of our problems.

In future we aim to apply the approach to realistic grid infrastructures (e.g., from the SciGRID project [22]) with the aim, to be able to quantify the technological feasibility of storage devices of systems with high shares of RES.

Acknowledgements CM acknowledges the support by the German Federal Ministry of Education and Research (BMBF) through the funding initiative “Zukunftsfähige Stromnetze” in the project SciGRID (funding code 03SF0471) and by the German Academic Exchange Service (DAAD) through the IPID4all programme. GM would like to thank CONICYT for financial support through Becas Chile (Number 72130388). DB, GM and SY acknowledge the support of DOE award “GMLC”.

References

- [1] Beaudin, M., Zareipour, H., Schellenberglabe, A., Rosehart, W.: Energy storage for mitigating the variability of renewable electricity sources: An updated review. *Energy for Sustainable Development* **14**(4), 302 – 314 (2010). DOI <http://dx.doi.org/10.1016/j.esd.2010.09.007>
- [2] Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: *Robust optimization*. Princeton University Press (2009)
- [3] Benders, J.: Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4**(1), 238–252 (1962)
- [4] Bertsimas, D., Brown, D.B., Caramanis, C.: *Theory and applications of robust optimization*. *SIAM review* **53**(3), 464–501 (2011)
- [5] Bienstock, D.: Histogram models for robust portfolio optimization. *J. Comput. Finance* **11**, 1 – 64 (2007)
- [6] Bienstock, D.: *Electrical Transmission System Cascades and Vulnerability: An Operations Research Viewpoint*, vol. 22. SIAM (2016)
- [7] Bienstock, D., Chertkov, M., Harnett, S.: Chance-constrained DC-OPF. *SIAM Review* **56**, 461–495 (2014)
- [8] Bienstock, D., Mattia, S.: Using mixed-integer programming to solve power grid blackout problems. *Discrete Optimization* **4**, 115 – 141 (2007)
- [9] Bienstock, D., Özbay, N.: Computing robust basestock levels. *Discrete Optimization* **5**, 389 – 414 (2008)
- [10] Bienstock, D., Verma, A.: The N - k Problem in Power Grids: New Models, Formulations and Numerical Experiments. *SIAM J. Optimization* **20**, 2352–2380 (2010)
- [11] Bludszweit, H., Domínguez-Navarro, J.A.: A probabilistic method for energy storage sizing based on wind power forecast uncertainty. *IEEE Transactions on Power Systems* **26**(3), 1651–1658 (2011)
- [12] Bludszweit, H., Domínguez-Navarro, J.A., Llombart, A.: Statistical analysis of wind power forecast error. *IEEE Transactions on Power Systems* **23**(3), 983–991 (2008)

- [13] Brown, P.D., Lopes, J.P., Matos, M.A.: Optimization of pumped storage capacity in an isolated power system with large renewable penetration. *IEEE Transactions on Power systems* **23**(2), 523–531 (2008)
- [14] Dufó-López, R., Bernal-Agustín, J.L., Contreras, J.: Optimization of control strategies for stand-alone renewable energy systems with hydrogen storage. *Renewable energy* **32**(7), 1102–1126 (2007)
- [15] Glover, J., Sarma, M., Overbye, T.: *Power System Analysis and Design*. Cengage Learning (2011)
- [16] Hedayati, M., Zhang, J., Hedman, K.: Joint transmission expansion planning and energy storage placement in smart grid towards efficient integration of renewable energy (2014)
- [17] Houston, E., Kennelly, A.: The electric motor and the transmission power. *Elementary electro-technical series*. The W. J. Johnston Company (1896)
- [18] Jannati, M., Hosseinian, S., Vahidi, B., Li, G.J.: A survey on energy storage resources configurations in order to propose an optimum configuration for smoothing fluctuations of future large wind power plants. *Renewable and Sustainable Energy Reviews* **29**, 158–172 (2014)
- [19] Jiang, R., Wang, J., Guan, Y.: Robust unit commitment with wind power and pumped storage hydro. *IEEE Transactions on Power Systems* **27**(2), 800–810 (2012)
- [20] Lorca, A., Sun, X.: Multistage robust unit commitment with dynamic uncertainty sets and energy storage. *IEEE Transactions on Power Systems* **PP**(99), 1–1 (2016)
- [21] Luo, X., Wang, J., Dooner, M., Clarke, J.: Overview of current development in electrical energy storage technologies and the application potential in power system operation. *Applied Energy* **137**, 511 – 536 (2015). DOI <http://dx.doi.org/10.1016/j.apenergy.2014.09.081>
- [22] Matke, C., Medjroubi, W., Kleinhans, D.: SciGRID - An Open Source Reference Model for the European Transmission Network (v0.2) (2016). URL <http://www.scigrid.de>
- [23] Oldham, K.: The Doctrine of Description: Gustav Kirchhoff, Classical Physics, and the “purpose of All Science” in 19th-century Germany. University of California, Berkeley (2008)
- [24] Qiu, T., Xu, B., Wang, Y., Dvorkin, Y., Kirschen, D.: Stochastic multi-stage co-planning of transmission expansion and energy storage. *IEEE Transactions on Power Systems* **PP**(99), 1–1 (2016). DOI 10.1109/TPWRS.2016.2553678
- [25] Schavemaker, P., van der Sluis, L.: *Electrical Power System Essentials*. Wiley (2008)
- [26] Siemer, L., Schöpfer, F., Kleinhans, D.: Cost-optimal operation of energy storage units: Benefits of a problem-specific approach. *Journal of Energy Storage* **6**, 11–21 (2016)
- [27] Wu, J., Zhang, B., Li, H., Li, Z., Chen, Y., Miao, X.: Statistical distribution for wind power forecast error and its application to determine optimal size of energy storage system. *International Journal of Electrical Power & Energy Systems* **55**, 100–107 (2014)
- [28] Zimmerman, R.D., Murillo-Sanchez, C.E.: *Matpower 6.01b User’s Manual*. Power Systems Engineering Research Center (Pserc) (2016). URL <http://www.pserc.cornell.edu/matpower/manual.pdf>
- [29] Zimmerman, R.D., Murillo-Sanchez, C.E., Thomas, R.J.: MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education. *IEEE Transactions on Power Systems* **26**(1), 12–19 (2011). DOI 10.1109/TPWRS.2010.2051168

An Image Segmentation Algorithm based on Community Detection

Youssef Mourchid, Mohammed El Hassouni and Hocine Cherifi

Abstract With the recent advances in complex networks, image segmentation becomes one of the most appropriate application areas. In this context, we propose in this paper a new perspective of image segmentation by applying two efficient community detection algorithms. By considering regions as communities, these methods can give an over-segmented image that has many small regions. So, the proposed algorithms are improved to automatically merge those neighboring regions agglomerative to achieve the highest modularity/stability. To produce sizable regions and detect homogeneous communities, we use the combination of a feature based on the Histogram of Oriented Gradients of the image, and feature based on color to characterize the similarity of two regions. By constructing the similarity matrix in an adaptive manner, we avoid the problem of the over-segmentation. We evaluate the proposed algorithms for Berkeley Segmentation Dataset, and we show that our experimental results can outperform other segmentation methods in terms of accuracy and can achieve much better segmentation results.

Keywords: Image segmentation; complex networks; community detection; modularity

1 Introduction

Image segmentation is still a challenging issue in visual information processing. Its goal is to split the image into homogeneous regions that represent similar features. It constitutes an essential issue in pattern recognition due to its practical importance. For example image, segmentation procedures in medical imaging can be used for diagnosis, allowing locating tumors and other pathologies [1]. Also, image segmentation techniques can be applied to machine vision, localization of objects in satellite images, and traffic control systems [2]. In recent years, graphs have emerged as a representation for image analysis and processing. Many powerful algorithms in image processing have been

Youssef Mourchid (e-mail: youssefmour@gmail.com) · Mohammed El Hassouni (e-mail: mohamed.elhassouni@gmail.com)
LRIT, URAC No 29, Faculty of Sciences, Mohammed V University in Rabat, B.P.1014 RP, Rabat, Morocco.

Mohammed El Hassouni
DESTEC, FLSHR, Mohammed V University in Rabat, Rabat, Morocco.

Hocine Cherifi (e-mail: hocine.cherifi@u-bourgogne.fr)
LE2I UMR 6306 CNRS, University of Burgundy, Dijon, France

formulated on graphs, i.e., a pixel in an image is the vertex in the graph, and the edge is determined by an adjacency relation among the image pixels. Using graphs in the image is not absolutely a new idea and there are many published methods of graph similarity testing. The common idea of all these methods is the construction of a weighted graph, where each vertex corresponds to an image pixel or a region, and the weight of each edge connecting two vertices represents the similarity that they belong to the same segment. Several key factors affect image segmentation, for example, proximity, similarity, regularity, i.e., the repetitive patterns, relative size and etc. In this paper, we will take into consideration all these factors. A lot of image segmentation algorithms have been proposed in the literature: *Region Based* [3], *Watershed* [4]-[7], *Feature based Clustering* [8] and *Mean Shift* algorithm [9].

Inspired by the application of community detection algorithms in large networks, we try to view an image as a network or a graph. For a network, modularity [10] and stability [11] are crucial quantities, which are used to evaluate the performance of community detection algorithms when the underlying community structure is not known. Unlike the existing image segmentation algorithms, the proposed approach identifies the differences between community detection and image segmentation and, proposes a texture feature to count the occurrences of gradient orientation in localized portions and encode it into a similarity matrix. The similarity among regions of pixels is constructed in an adaptive manner for avoiding the over-segmentation. The proposed algorithms can automatically detect the number of regions in an image compared with other existing segmentation algorithms, it can also produce sizable regions and achieves much better semantic level segmentation of the image. The proposed contributions of this paper are the following:

- Efficient community detection algorithms are used. They present a low time complexity as well as comparable performance.
- A texture feature named Histogram of Oriented Gradients (HOG) [12] is used to detect regions of interest in the image. The HOG feature, together with the color feature, encodes much better similarity measure from the semantic point of view.
- Finally, the construction of an adaptive similarity matrix W is proposed to avoid the over-segmentation. At each iteration, the similarity between two regions of the image is recalculated. The goal is to avoid breaking visually coherent regions, which have smooth changes in color or texture caused by shadow or perspectives.

The rest of the paper is organized as follows. In Section 2, we introduce briefly the concept of community detection and modularity/stability. Then, we recall efficient community detection algorithms. Section 3 explains how image segmentation and community detection can be related followed by the description of the proposed contribution and the detailed technical points. Experiments on the publicly Berkeley Segmentation Data Set (BSDS500) are reported in Section 4. Finally, in Section 5, we present our conclusions.

2 Community Detection

Community Detection is a hot topic in network science during the past few years, and it's a very prolific subject in the complex network literature [13]. A community is a group of nodes which are densely connected with each other and are sparsely connected with members of other communities. The community detection is a fundamental problem, which objective is to find the best division of the network into their constituent communities. Several algorithms have been developed so far to deal with this issue. Numerous solutions to solve this problem, are linked to a measure called modularity. Introduced by Newman [14], it measures the quality of a community structure and it is defined as follow:

$$Q = \Sigma(e_{ii} - a_i^2) \quad (1)$$

Where e_{ii} denotes the fraction of network edges which are inserted into a community i , and a_i^2 denotes the fraction considering that edges are inserted randomly. The modularity value Q is between 0 and 1. A high value of the modularity indicates a strong community structure of the network.

Another quality measure called stability Q_s was introduced in [11] based on the clustered auto-covariance of a dynamic Markov process. It measures the quality of a partition as a community structure. Because the stability has an intrinsic dependence on time scales of the graph, it can allow the comparison and the ranking of the partitions at each time and also establish the time spans over which partitions are good and optimal. Thus, the Markov time acts effectively as an intrinsic resolution parameter that establishes a hierarchy of increasingly coarser communities.

Several algorithms have been developed to find a partition of the network which is a good approximation of maximum modularity or stability. In the following, we present two influential algorithms that we propose to use.

2.1 Fast multi-scale detection of communities using stability optimization

Modularity initially was introduced to evaluate the quality of partitions. Nevertheless, its use has broadened from partition quality measure to optimization function and now modularity optimization is a very common technique to detect communities. In this algorithm [16], the variation in modularity ΔQ_M to merge two communities i and j is computed as follows:

$$\Delta Q_{M_{ij}} = 2(e_{ij} - a_i a_j) \quad (2)$$

Where i and j denote the merged communities in the new candidate partition. When a better dQ is found when moving a node, the algorithm checks that moving this node does not leave its initial community disconnected. Otherwise, some communities may end up being in several components that should not be grouped together as one. Note that this implementation uses two distinct lists of neighbors. The first lists the actual neighbors in the initial network and the second stores the neighbors in the current matrix for the given parameter. This is necessary in order to only consider actual neighbors when selecting the candidate neighbor nodes and communities. The computation of the matrices for each parameter value is optimized by keeping in memory the recent matrices and corresponding exponents. For each new exponent, the optimization attempts to exploit the previously computed matrices to speed up the matrix power computation.

2.2 Modularity optimization based on Danon greedy agglomerative method

It's a greedy community detection agglomerative method which has been introduced by Danon [17]. The algorithm process is a simple modification of the algorithm proposed by Newman for detecting communities. The greedy method of Newman is an agglomerative hierarchical clustering method, where groups of nodes are successively joined to form large communities such that the value of modularity increases after this merging. This greedy optimization of modularity tends to form faster large communities at the expenses of small ones, which often yields poor values of the maximum value of modularity. Danon suggested a normalization of the modularity variation produced by merging two communities by the fraction of edges incident to one of the two communities, in order to avoid having small communities. This trick leads to better modularity value as compared to the original recipe of Newman, especially when communities have different sizes

3 Segmentation algorithms

Due to the inherent properties of images, there is a difference between the segmentation and community detection problems, and applying directly community detection algorithms to image segmentation [18], by considering the pixels as nodes of the network, lead to low performance. The following aspect reveals the difference between image segmentation and community detection. First, pixels in segmentation possibly have completely different properties, like the color but in community detection, they share similar properties. Second, a single pixel cannot capture regularities and information in each visually homogeneous segment of the image. Third, images share some information compared with communities, for example, adjacent regions are more likely to belong to the same segment.

So, to solve the mentioned problems, we propose an approach which takes advantage of the efficient optimization in modularity/stability using community detection algorithms and also the inherent properties of the image. The proposed algorithms start by an initial segmentation which split the image into homogeneous regions, possibly small regions which are used in the next steps. The proposed segmentation approach is explained in Algorithm.1 and the detail of some technical points are defined below.

Algorithm 17

Input: Given a color image I and its over-segmented initialization with a set of super-pixels $R = \{R_1, \dots, R_n\}$ where n is the number of super-pixels

Output: The set of image segments $C_i = \{C_{i1}, \dots, C_{ic}\}$ with $c \leq n$

- 1: **while** community structure still change ($C_i \neq C_{i-1}$) **do**
 - 2: Construct the neighborhood system for each region R_i ;
 - 3: Compute the Histogram of Oriented Gradients texture feature and estimate the distribution of the color feature for each region R_i ;
 - 4: Adaptively update the similarity matrix W according to Equation (5), $w_{ij} \neq 0$ only if R_i and R_j are adjacent regions in I
 - 5: **while** modularity/stability increases by merging any two adjacent regions **do**
 - 6: Compute the community structure using a community detection algorithm $C_i = \{C_{i1}, \dots, C_{im}\}$ where m is the current number of communities
 - 7: **end while**
 - 8: **end while**
-

3.1 Super-pixels

In this paper, the super-pixels are chosen as an initial segmentation because we want to avoid the over-segmentation issue. It is characterized by the splitting of the same perceptual region in a multitude of smaller regions. Furthermore, super-pixels can well preserve the object boundaries. The proposed community detection algorithms start the process of aggregation by treating each single pixel as a community which will take more time and also there is no reason to treat a single pixel because it contains no information about texture. For these reasons, we start with an initial segmentation by super-pixels which are a set of very small regions of pixels. Using this pre-segmentation can greatly reduce the complexity without affecting the segmentation performance. In this paper, we use a publicly available code [25] to get the initial segmentation. As shown in Figure 1, the initial segmentation by the super-pixel generation step gives more than 200 over

segmented regions which can greatly reduce the complexity by considering only 200 nodes instead of a large number of nodes in the first iteration for the proposed algorithms.

3.2 Construction of the similarity matrix

Images have self-contained spatial a priori information which is used to construct different neighborhood system. For more specification, we consider the possibility of merging neighboring regions in the image by considering the adjacent regions of each region in the image to be its neighbors and store its neighboring regions using an adjacent list which contains the regions that share at least one pixel with the current region.

3.2.1 Features to compute the similarity

The most straightforward and important feature for segmentation is color. Various color spaces are proposed in the literature to capture different aspects of the color, such as L^*a^*b , HSV, YUV, and RGB. The choice of an appropriate color space is a very important step for achieving a good segmentation performance. We choose the L^*a^*b color space because it's known to be in accordance with the human visual system and perceptually uniform. It is a 3-axis color system with dimension L for lightness and a and b for the color dimensions.

We use the pixel value in the L^*a^*b color space as a feature to compute the similarity between regions. Nevertheless, using this feature only cannot achieve good segmentation performance, because in some homogeneous object using just the color feature will break down image regularities into different segments. To solve this problem, we propose to use a texture feature called Histogram of Oriented Gradients (HOG). HOG is a feature descriptor used in computer vision and image processing to detect objects. This technique counts occurrences of gradient orientation in localized portions of an image detection window, or region of interest. We construct the Histogram of Oriented Gradients as follows:

- We divide the image into small connected regions (cells). For each cell, we compute a histogram of gradient directions or edge orientations for the pixels within the cell.
- We discretize each cell into angular bins according to the gradient orientation.
- Each cell's pixel contributes weighted gradient to its corresponding angular bin.
- Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.
- Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor.

3.2.2 Similarity measure

Different similarity measures are used for the two features. For the color feature, a three-dimensional vector in the L^*a^*b color space represents each pixel in the image. For measuring the similarity between two regions in the image, the pixel value in the same region is represented by a three-dimensional Gaussian distribution. Several distance measures for distributions are studied in the literature like Kullback-Leibler (KL) Divergence, Mean Distance, and Earth Mover's Distance. In the proposed algorithms, to compute the distance between the two color feature distributions for two regions of pixels, we use the Mean Distance (MD). We use a Gaussian type radius basis function for transforming the above distribution distance into similarity measure defined by:

$$c_{ij}(color) = \exp\left\{\frac{-dist(R_i, R_j)}{2\sigma}\right\} \quad (3)$$

Where $dist(R_i, R_j)$ denotes the distance between the pixel value distributions for region R_i and R_j .

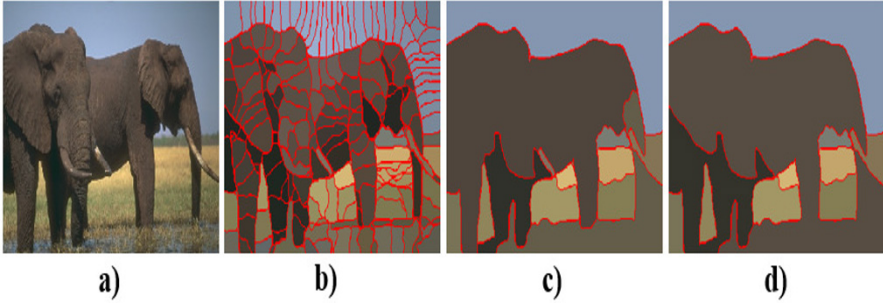


Fig. 1: a) Original image; b) Super-pixels image; c) Fast multi-scale using stability optimization; d) Modularity optimization based on Danon;

For the proposed Histogram of Oriented Gradients (HOG), we use the cosine similarity to measure the similarity between the regions where each region is represented by a 256 dimensional vector and for two regions R_i and R_j . The HOG feature vector is $h_i, h_j \in R^{256}$ as indicated by:

$$t_{ij}(\text{texture}) = \cos(h_i, h_j) = \frac{h_i^T h_j}{\|h_i\| \cdot \|h_j\|} \quad (4)$$

3.2.3 Construction of the adaptive similarity matrix

The process of constructing the similarity matrix W is adaptive. During each iteration, we maintain an adaptive similarity matrix by recomputing the similarity between each two regions again in accordance with the equation (3) and (4). The reason for using this process is because, during the aggregation process of the community detection algorithms, regions keep expanding. The similarity measure resulting from the previous iteration might not suitable for the current iteration. So, using an adaptive similarity matrix reevaluate the similarity between current regions. It avoids over-segmentation and finally overcomes the problem of splitting the non-uniformly distributed color or texture, which should be grouped into the same community in the image from the perspective of the human visual system. To construct the adaptive similarity matrix W during each iteration, we use a hybrid model to combine the color feature and the HOG texture feature as defined below:

$$W = w_{ij} = a \times \sqrt{t_{ij}(\text{texture}) \times c_{ij}(\text{color})} + (1 - a) \times c_{ij}(\text{color}); (i, j) = 1, \dots, n \quad (5)$$

Where n is the number of regions and a denotes a balancing parameter. If the texture information is not taken into consideration, i.e., $a = 0$, the more we increase the value of a , the more stripe patterns are encoded into the similarity, thus better preserves the regularities and information in the image. Nevertheless, if a is too large, some distinct objects in the image are merged into one segment. In our experiments, we give higher priority to the color feature.

4 Experiments and results

This section provides experiments that were conducted to assess the performance of the proposed approach qualitatively as well as quantitatively. The proposed algorithms are tested on the publicly available Berkeley Segmentation Data Set 500 (BSDS500) [19]. BSDS500 contains 100 validation images of size 321×481 pixels that are randomly chosen from the Corel database. These images are manually segmented by humans in a natural way. In the qualitative evaluations experiment,

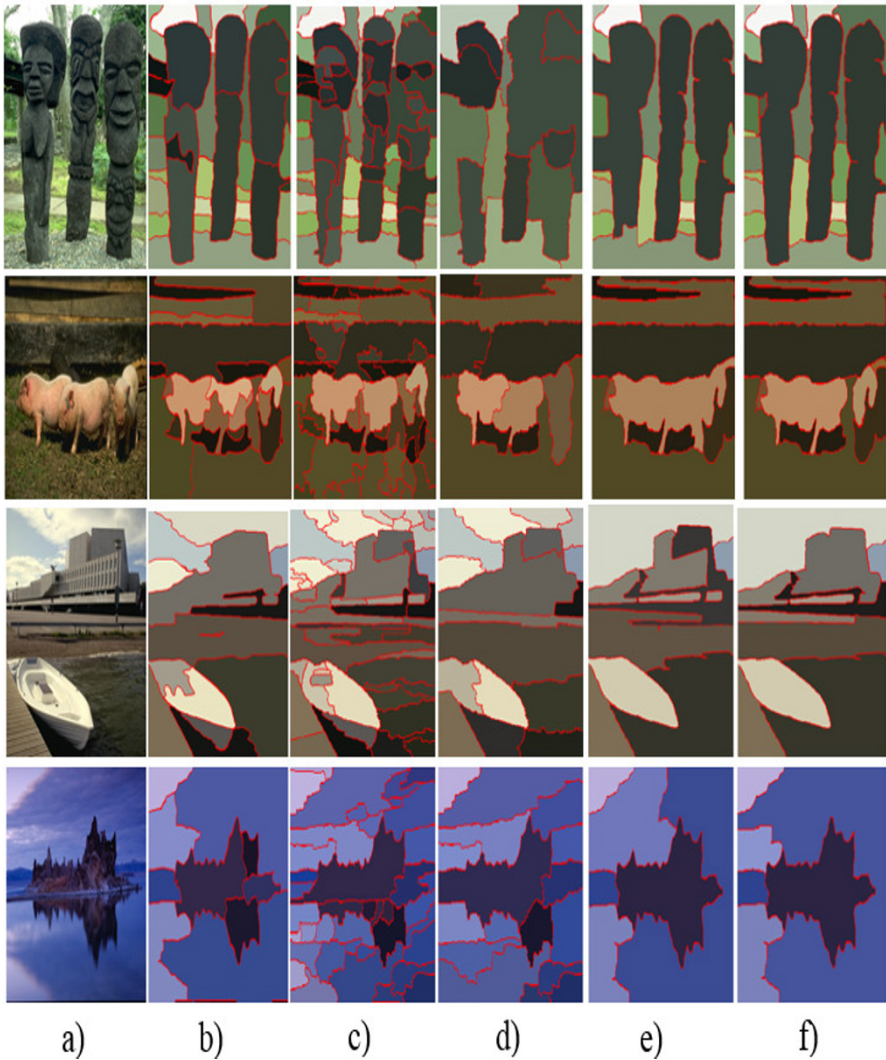


Fig. 2: a) Original image; b) LC; c) JSEG; d) EDISON; e) Fast multi-scale using stability optimization; f) Modularity optimization based on Danon;

figure 1 shows the results of the proposed algorithms with the adaptive similarity matrix for image segmentation. So, we can see that the proposed algorithms give much better results and produce sizable regions for all selected images. Even if some pixel in the image have different values inside the same regions, the adaptive similarity matrix of HOG texture feature can successfully preserve the regularities and classifies those pixels into the same segment.

We have performed a qualitative and quantitative comparisons of the proposed algorithms based on the adaptive similarity matrix with some existing state of the art segmentation methods: Lossy Compression (LC) [20], EDISON [21] and JSEG [22]. As shown in figure 2, LC, EDISON, and

JSEG show the different extent of over-segmentation and break the regularities in some homogeneous regions of the image compared to the proposed approach which preserves the regularities and produces sizable homogeneous regions.

For the quantitative evaluation, we evaluate the segmentation performance of the proposed algorithms with the three segmentation techniques. We investigate for the quantitative evaluation the Probabilistic Rand Index [23] which is a classical evaluation criteria for clustering. It measures the probability that the pair of samples has consistent labels in the two segmentations. A larger value indicates a greater similarity between two segmentations. The range of PRI is [0,1]. Table1 presents the average values of the PRI, which were applied to all of the 100 images in the Berkeley segmentation dataset. Again, it has been observed that the proposed algorithms work better for the image segmentation task among all the popular segmentation algorithms LC, EDISON and JSEG in term of PRI and have a close performance to human perception.

Algorithms	PRI
Humain	0.870
Fast multi-scale	0.811
Modularity optimization based on Danon	0.803
EDISON	0.786
JSEG	0.760
LC	0.778

Table 1: Quantitative comparison of different algorithms on Berkeley dataset

5 Conclusion

This paper proposed an efficient image segmentation algorithm taking advantages of the optimization of modularity/stability and the inherent properties of images. To optimize modularity/stability, we used the efficient community detection algorithms, Fast multi-scale using stability optimization and Modularity optimization based on Danon which can automatically detect the number of segments in the image. By employing the color feature and the Histogram of Oriented Gradients (HOG) texture feature, we constructed the similarity matrix adaptively among different regions by optimizing the modularity/stability and aggregated the neighboring regions iteratively. When no modularity/stability increase occurs by aggregating any neighboring regions, the optimal segmentation is achieved. Results of our experiments have proved that the proposed algorithms give an impressive qualitative segmentation result as shown in the figures and achieve the best performance quantitatively among all the experimented popular methods in terms of PRI. Since, using two efficient community detection algorithms, the proposed approach avoid the over-segmentation problem and preserve the regularities in the object.

References

- [1] J. A. Noble and D. Boukerroui, "Ultrasound image segmentation: A survey," IEEE Transactions on medical imaging, 25(8), 987-1010.
- [2] H. Zhang, J. E. Fritts, and S. A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," Computer Vision and Image Understanding, Elsevier, vol. 110, no. 2, pp. 260-280, 2008.
- [3] A. M. Khan, Ravi. S., "Image Segmentation Methods: A Comparative Study", IJSCE, Volume-3, Issue-4, September 2013.

- [4] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583-598, 1991.
- [5] V. Grau, A. Mewes, M. Alcaniz, R. Kikinis, and S. K. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 447-458, 2004.
- [6] X.-C. Tai, E. Hodneland, J. Weickert, N. V. Bukoreshtliev, A. Lundervold, and H.-H. Gerdes, "Level set methods for watershed image segmentation," in *Scale Space and Variational Methods in Computer Vision*. Springer, 2007, pp. 178-190.
- [7] V. Osma-Ruiz, J. I. Godino-Llorente, N. Saenz-Lech on, and P. G omez-Vilda, "An improved watershed algorithm based on efficient computation of shortest paths," *Pattern Recognition*, vol. 40, no. 3, pp. 1078-1090, 2007.
- [8] R.Yogamangalam, B.Karthikeyan, "Segmentation Techniques Comparison in Image Processing", *IJET*, Vol 5, No 1, Feb-Mar 2013.
- [9] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1271-1283, 2010.
- [10] L. Angelini, D. Marinazzo, M. Pellicoro, and S. Stramaglia, "Natural clustering: the modularity approach," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2007, no. 08, p. L08001, 2007.
- [11] J.-C. Delvenne, S. N. Yaliraki, and M. Barahona, "Stability of graph communities across time scales," *Proceedings of the National Academy of Sciences*, vol. 107, no. 29, pp. 12755-12760, 2010.
- [12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR.*, vol. 1. IEEE, 2005, pp. 886-893.
- [13] M. A. Porter, J.-P. Onnela, and P. J. Mucha, "Communities in networks," *Notices of the AMS*, vol. 56, no. 9, pp. 1082-1097, 2009.
- [14] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.
- [15] M. E. Newman, "Analysis of weighted networks," *Physical Review E*, vol. 70, no. 5, p. 056131, 2004.
- [16] Le Martelot, Erwan, and Chris Hankin. "Multi-scale community detection using stability optimisation within greedy algorithms." *arXiv preprint arXiv:1201.3307*(2012).
- [17] L. Danon, A. Diaz-Guilera, A. Arenas, "The effect of size heterogeneity on community identification in complex networks" , *J. Stat. Mech.* 11 (2006).
- [18] Mourchid, Y., El Hassouni, M., and Cherifi, H. "Image segmentation based on community detection approach". *The International Journal of Computer Information Systems and Industrial Management Applications*. ISSN 2150-7988 Volume 8 (2016) pp. 195-204
- [19] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898-916, 2011.
- [20] A. Yang, J. Wright, Y. Ma, and S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212-225, 2008.
- [21] C. M. Christoudias, B. Georgescu, and P. Meer, "Synergism in low level vision," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 4. IEEE, 2002, pp. 150-155.
- [22] Y. Deng and B. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 8, pp. 800-810, 2001.
- [23] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 929-944, 2007

- [24] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 929-944, 2007
- [25] G. Mori, "Guiding model search using segmentation," in *The Proceedings of the 10th IEEE International Conference on Computer Vision, ICCV.*, vol. 2. IEEE, 2005, pp. 1417-1423.

Erratum to: Identifying Influential Spreaders by Graph Sampling

Nikos Salamanos, Elli Voudigari and Emmanuel J. Yannakoudakis

Erratum to:
Chapter “Identifying Influential Spreaders by Graph Sampling” in: H. Cherifi et al. (eds.), *Complex Networks & Their Applications V*, Studies in Computational Intelligence 693, DOI [10.1007/978-3-319-50901-3_9](https://doi.org/10.1007/978-3-319-50901-3_9)

The original version of the book was inadvertently published without the following corrections:

1. The Figs. 2–4 wrongly placed after the paper references should be placed before Sect. 5 Conclusion in pages 118–120.
2. The “Algorithm-3” in pages 113 & 114 should be changed to read as “Algorithm-1”.
3. In page 113, after Step-10, the numbering of the algorithmic steps should be corrected to match the original version and inside Algorithm, Steps-2 and 13 have wrong equation alignment to be corrected.
4. In page 116, Table 1, first row, second column: spacing problem in the words “egoFacebook” and “wiki-Vote” should be cleared.
5. Almost all sub-figure links leading to other papers of the proceedings should be changed to lead to correct figures of the paper.
Specifically, the links, in Sect. 4.3:
Fig. 1a, page-116, line-14
Fig. 1b and 1c, page-116, line-14
Fig. 2a, page-116, line-19
Fig. 2b, page-116, line-21

The updated original online version for this chapter can be found at http://dx.doi.org/10.1007/978-3-319-50901-3_9

Fig. 2c, page-116, line-22

Fig. 3a and 3b, page-117, line-12

Fig. 3d, page-118, line-3

Author Index

A

Abba, Sofiane, 551
Agarwal, Shivam, 579
Akin, Myles, 697
Alsulaiman, Talal, 671
Antoniuk, Artem, 785
Arnaboldi, Valerio, 459
Arrigo, Francesca, 147

B

Baffier, Jean-François, 159
Baggag, Abdelkader, 551
Bahulkar, Ashwin, 423
Basov, Nikita, 785
Berthouze, Luc, 223
Bessi, Alessandro, 595
Bienstock, Daniel, 809
Bockholt, Mareike, 183
Borge-Holthoefer, Javier, 551
Bottone, Michele, 361
Boukerram, Abdellah, 335
Box-Steffensmeier, Janet M., 349
Braun, Fabian, 721
Buzzanca, Marco, 299

C

Caelen, Olivier, 721
Caldarelli, Guido, 595
Canu, Maël, 275
Carchiolo, Vincenza, 299
Casamassima, Francesca, 539
Chan, Kevin, 423
Cherifi, Hocine, 821
Christenson, Dino P., 349
Claver, Vivek, 159
Cottica, Alberto, 41
Cox, Scott, 437

Crawford, Brian, 209
Cremonini, Marco, 539

D

d'Allonnes, Adrien Revault, 275
de Barros Pereira, Hernane Borges, 321
Debnath, Joyati, 197
de Freitas Piedade Melo, Dirceu, 321
Del Vicario, Michela, 595
de Mello Araújo, Eric Fernandes, 773
de Sousa Fadigas, Inacio, 321
Dessi, Danilo, 709
Di Francesco Maesa, Damiano, 749
Di Nardo, Armando, 797
Di Natale, Michele, 797
Drif, Ahlem, 335
Du, Siying, 373
Dzakpasu, Rhonda, 697

E

El Hassouni, Mohammed, 821
Espin-Noboa, Lisette, 3

F

Fenu, Gianni, 709
Figueira, Alvaro, 631
Fortuna, Paula, 631

G

Garg, Kamini, 459
Gera, Raluca, 209
Ghosh, Supratim, 525
Giordano, Silvia, 335, 459
Giudicianni, Carlo, 797
Greco, Roberto, 797
Gregory, Steve, 373
Grimm, Alexander, 171

Gueye, Ibrahima, 263
 Guo, Yixin, 697
 Gutfrand, Alexander, 17

H

Hamann, Michael, 17
 Hasani-Mavriqi, Ilire, 385
 Head, B., 487
 Hecking, Tobias, 123
 Hegde, Kshiteesh, 287
 Helic, Denis, 247, 385
 Higham, Desmond J., 147
 Hillebrand, A., 685
 Hmimida, Manel, 309
 Hoppe, H.Ulrich, 123
 Horadam, K.J., 437
 House, Jeffrey, 209

I

Iervolino, Raffaele, 397
 Ikeda, Yuichi, 657

J

Jansen, Marc, 123

K

Kanawati, Rushed, 309
 Khairuddin, Mohammad Adib, 619
 Khashanah, Khaldoun, 671
 Kheddouci, Hamamache, 473
 Kiss, Istvan Z., 223
 Klein, Michel, 773
 Kleinhans, David, 809
 Knuth, Thomas, 209
 Kumari, Suchi, 29
 Kuzmin, Konstantin, 287

L

Lalou, Mohammed, 473
 Lamprecht, Daniel, 247
 Larson, Jennifer M., 449
 Lebichot, Bertrand, 721
 Lee, Ju-Sung, 785
 Lemmerich, Florian, 3
 Lerner, Jürgen, 95
 Lesot, Marie-Jeanne, 275
 Lex, Elisabeth, 385
 Li, Cheng-Te, 735
 Liebig, Jessica, 619
 Lin, Jianyi, 235
 Liu, Qiang, 511
 Lizardo, Omar, 423
 Lomi, Alessandro, 95

Longheu, Alessandro, 299
 López, Julio César Amador Díaz, 607
 Lotfi, Dounia, 197

M

Magdon-Ismail, Malik, 287
 Malgeri, Michele, 299
 Mangioni, Giuseppe, 299
 Marik, Radek, 567
 Marino, Andrea, 749
 Marraki, Mohamed El, 197
 Märtens, M., 685
 Matke, Carsten, 809
 Meier, J., 685
 Melançon, Guy, 41
 Mesiti, Marco, 235
 Meyerhenke, Henning, 17
 Miller, Ryan, 209
 Mokhlissi, Raihana, 197
 Mourchid, Youssef, 821
 Muñoz, Gonzalo, 809

N

Ndong, Joseph, 263
 Nefedov, Nikolai, 761

O

Oliveira, Luciana, 631
 Onderdonk, Alex, 697
 Osaka, Kengo, 411
 Overbury, Peter, 223

P

Pang, Jun, 735
 Parek, Deven, 525
 Pau, Pier Luigi, 709
 Peperkamp, Sharon, 83
 Piña-García, C.A., 607
 Primiero, Giuseppe, 361

Q

Qasem, Ziyaad, 123
 Qu, Bo, 499

R

Raimondi, Franco, 361
 Rao, Asha, 437, 619
 Re, Matteo, 235
 Renoust, Benjamin, 41, 159
 Ricci, Laura, 749
 Roelofsma, Peter, 55
 Ruths, Derek, 525
 Ruths, Justin, 525

S

Saerens, Marco, [721](#)
Safro, Ilya, [17](#)
Sager, Sebastian, [809](#)
Salamanos, Nikos, [111](#)
Sandim, Miguel, [631](#)
Santonastaso, Giovanni Francesco, [797](#)
Singer, Philipp, [3](#)
Singh, Anurag, [29](#)
Slimani, Yacine, [335](#)
Spelta, Alessandro, [645](#)
Sreevalsan-Nair, Jaya, [579](#)
Srivastava, Jaideep, [551](#)
Stanisavljevic, Darko, [385](#)
Staudt, Christian L., [17](#)
Strohmaier, Markus, [3](#), [247](#), [385](#)
Sugawara, Toshiharu, [411](#)
Szymanski, Boleslaw K., [287](#), [423](#)

T

Tagliabue, Jacopo, [361](#)
Tangredi, Domenico, [397](#)
Tavassoli, Sude, [135](#)
Tessone, Claudio J., [171](#)
Tewarie, P., [685](#)
Thomas, Jijju, [525](#)
Tomar, Amit, [579](#)
Toriumi, Fujio, [411](#)
Treur, Jan, [55](#), [69](#)

Turnbull, Rory, [83](#)

V

Valentini, Giorgio, [235](#)
van Halteren, Aart, [773](#)
van Ments, Laila, [55](#)
Van Mieghem, P., [685](#)
Van Mieghem, Piet, [511](#)
Vasca, Francesco, [397](#)
Vermeer, W., [487](#)
Voudigari, Elli, [111](#)

W

Wang, Huijuan, [499](#)
Watanabe, Tsutomu, [657](#)
Wilensky, U., [487](#)

Y

Yang, Shuoguang, [809](#)
Yannakoudakis, Emmanuel J., [111](#)

Z

Zanouda, Tahar, [551](#)
Zhang, Qian, [595](#)
Zhang, Yang, [735](#)
Zhou, Lu, [735](#)
Zollo, Fabiana, [595](#)
Zweig, Katharina A., [135](#), [183](#)