

Marcello Pelillo · Irene Poli
Andrea Roli · Roberto Serra
Debora Slanzi · Marco Villani (Eds.)

Communications in Computer and Information Science

830

Artificial Life and Evolutionary Computation

12th Italian Workshop, WIVACE 2017
Venice, Italy, September 19–21, 2017
Revised Selected Papers

Communications in Computer and Information Science

830

Commenced Publication in 2007

Founding and Former Series Editors:

Alfredo Cuzzocrea, Xiaoyong Du, Orhun Kara, Ting Liu, Dominik Ślęzak,
and Xiaokang Yang

Editorial Board

Simone Diniz Junqueira Barbosa

*Pontifical Catholic University of Rio de Janeiro (PUC-Rio),
Rio de Janeiro, Brazil*

Phoebe Chen

La Trobe University, Melbourne, Australia

Joaquim Filipe

Polytechnic Institute of Setúbal, Setúbal, Portugal

Igor Kotenko

*St. Petersburg Institute for Informatics and Automation of the Russian
Academy of Sciences, St. Petersburg, Russia*

Krishna M. Sivalingam

Indian Institute of Technology Madras, Chennai, India

Takashi Washio

Osaka University, Osaka, Japan

Junsong Yuan

Nanyang Technological University, Singapore, Singapore

Lizhu Zhou

Tsinghua University, Beijing, China

More information about this series at <http://www.springer.com/series/7899>

Marcello Pelillo · Irene Poli
Andrea Roli · Roberto Serra
Debora Slanzi · Marco Villani (Eds.)


Artificial Life and Evolutionary Computation


12th Italian Workshop, WIVACE 2017
Venice, Italy, September 19–21, 2017
Revised Selected Papers

Editors


Marcello Pelillo 
Ca' Foscari University of Venice
Venice
Italy

Irene Poli 
Ca' Foscari University of Venice
Venice
Italy

Andrea Roli 
University of Bologna
Cesena
Italy

Roberto Serra 
University of Modena and Reggio Emilia
Modena
Italy

Debora Slanzi 
Ca' Foscari University of Venice
Venice
Italy

Marco Villani 
University of Modena and Reggio Emilia
Modena
Italy

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-319-78657-5 ISBN 978-3-319-78658-2 (eBook)
<https://doi.org/10.1007/978-3-319-78658-2>

Library of Congress Control Number: 2018937357

© Springer International Publishing AG, part of Springer Nature 2018

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. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The Wivace conference series has been showing a vitality that is quite surprising, as it is not supported by any formal society or association. Instead, it is the scientific community that gathers at Wivace that continues to make it so lively. At each conference there is an informal meeting, where the organizer of the following Wivace is chosen. This method provides a continuous testing of the value of the conference for the participants, and it also explains why the flavors of various editions may differ, depending in part on the main scientific orientation of the organizers. In 2017 the burden of the organization was mostly on mathematical modelers, computer scientists, and statisticians, while in the two previous editions it had been on biologists and chemists. However, the interest of the two communities for the other's work is always high, and no "parallel sessions" are run.

The dual soul of Wivace is indeed rooted in its history, since it was born out of the coalescence of two independent initiatives. Two workshops on Artificial Life were held in 2003 in Cosenza and in 2005 in Rome, and a workshop on Evolutionary Computation took place in Milan in 2005 (as a part of the Conference of the Italian Association for Artificial Intelligence). The organizers of the two initiatives decided to co-locate their 2006 editions in Siena, sharing a day (the last day of one conference coincided with the first day of the other). It became clear that there was a strong mutual interest between the two scientific communities, so it was decided that the two initiatives had to merge – and the beautiful acronym Wivace was born!¹

The first true Wivace took place in Samperi (Sicily) in 2007, followed by the 2008 edition in Venice. This was the first time that some invited speakers and some participants came from abroad, and the conference proceedings were published in English by an international publisher. Wivace 2009 took place in Naples, and in that case the local organizer decided to work with an Italian publisher, and to accept papers in Italian. This prompted some reflections about what Wivace had to be, and it was eventually decided that its international features had to be preserved and improved. Therefore, since the edition that took place in Parma in 2012, there have always been international proceedings and international well-known invited speakers.

A possibly incomplete list of non-Italian invited speakers include Stuart Kauffman, Norman Packard, Christian Mueller-Schloer, Wim Hordijk, Yaroslav Sergeyev, Ricard Solé, Kepa Ruiz Mirazo, Olli Yli-Harja, Gabor Vattay, Steen Rasmussen, Ruedi Fuechsli, and Erik Schultes. Moreover, several Italian researchers working abroad have also been invited to Wivace.

The meeting in Milan in 2013 was followed by Wivace 2014 in Vietri sul Mare, co-located with that year's edition of Wirm, a series of workshops on neural networks dating back to 1986. In 2015 the meeting took place in Bari and in 2016 in Salerno. And in 2017 we were back in Venice, for the ninth conference under the name of Wivace.

¹ In Italian, "vivace" means "lively."

However, since (as detailed above) three “parent” Wiva workshops had taken place earlier, this can be regarded as the 12th edition of the series.

Wivace is not a big conference, the number of participants typically ranging around 50. During the conference, presentations are discussed in depth, without, however, indulging in uselessly aggressive polemics. The combination of scientific rigor with a relaxed atmosphere, and with the openness to novel suggestions and hypotheses, also facilitates informal discussions and exchange of ideas.

We have been lucky to host some invited speakers who combined their outstanding scientific merits with the capability to effectively communicate their thoughts, and thus we are deeply indebted to Stuart Kauffman, Steen Rasmussen, Ruedi Fuechslin, Erik Schultes, and Roberto Taramelli for their contributions.

The review process was quite long, involving two phases. The call for papers left the choice of presenting an extended abstract or a full paper to the contributors, and the acceptance of the presentation to the conference was based on this document. Two reviewers were involved for each submission, and comments and suggestions were sent to the authors. After the conference, all the authors of accepted contributions were asked to send a full paper, which was scrutinized by three reviewers who also sent their evaluations and suggestions for improvements. The 23 papers in this volume are the outcome of this selection procedure. We thank all the contributors and all the participants for their role in making Wivace 2017 a successful event. And special thanks are due to the members of the Program Committee and the reviewers for their precious work.

This year’s organization profited from the strong and highly qualified support of the European Centre for Living Technology (ECLT), an international research center run by the Università Cà Foscari, which is associated with several universities and research institutions in Europe and the USA. We wish to thank in particular Agnese Boscarol and Roberta D’Argenio, who managed the complicated organizational and financial aspects of the conference, and Marco Fiorucci, who took care of the website. We also thank the student Feliks Hibraji for his help.

Thanks are also due to the host institutions of the organizers, namely, the Università Ca’ Foscari di Venezia, the Università di Modena e Reggio Emilia, and the Università di Bologna.

Let us finally acknowledge the precious advice of the staff at Springer, who provided their professional support through all the phases that led to this volume.

February 2018

Marcello Pelillo
Irene Poli
Roberto Serra
Andrea Roli
Debora Slanzi
Marco Villani

Organization

General Chairs

Marcello Pelillo	Ca'Foscari University of Venice, Italy
Irene Poli	Ca'Foscari University of Venice, Italy
Andrea Roli	University of Bologna, Italy
Roberto Serra	University of Modena and Reggio Emilia, Italy
Debora Slanzi	Ca'Foscari University of Venice, Italy
Marco Villani	University of Modena and Reggio Emilia, Italy

Program Committee

Michele Amoretti	University of Parma, Italy
Leonardo Bich	Universidad de Chile, Chile
Matteo Borrotti	CNR-IMATI, Italy
Michele Braccini	University of Bologna, Italy
Marcello Antonio Budroni	University of Sassari, Italy
Stefano Cagnoni	University of Parma, Italy
Angelo Cangelosi	University of Plymouth, UK
Timoteo Carletti	University of Namur, Belgium
Antonio Chella	University of Palermo, Italy
Simona Concilio	University of Salerno, Italy
Chiara Damiani	University of Milan Bicocca, Italy
Luca Di Gaspero	University of Udine, Italy
Alessandro Filisetti	Explora Biotech Srl, Italy
Francesco Fontanella	University of Cassino and Southern Lazio, Italy
Massimo Franceschet	University of Udine, Italy
Mario Giacobini	University of Turin, Italy
Alex Graudenzi	University of Milano-Bicocca, Italy
Roberto Marangoni	University of Pisa, Italy
Giancarlo Mauri	University of Milano-Bicocca, Italy
Sara Montagna	University of Bologna, Italy
Stefano Piotto	University of Salerno, Italy
Clara Pizzuti	CNR-ICAR, Italy
Riccardo Righi	European Commission, Joint research Center, Spain
Simone Righi	University College London, UK
Federico Rossi	University of Salerno, Italy
Hiroki Sayama	Binghamton University, SUNY, USA
Giandomenico Spezzano	CNR-ICAR and University of Calabria, Italy
Pasquale Stano	University of Salento, Italy
Thomas Stuetzle	Université Libre de Bruxelles, Belgium
Pietro Terna	University of Turin, Italy

Marco Tomassini	University of Lausanne, Switzerland
Vito Trianni	ISTC-CNR, Italy
Olli Yli-Harja	Tampere University of Technology, Finland

Local Organizing Committee

Agnese Boscarol	European Centre for Living Technology, Ca'Foscari University of Venice, Italy
Roberta d'Argenio	European Centre for Living Technology, Ca'Foscari University of Venice, Italy
Marco Fiorucci	European Centre for Living Technology, Ca'Foscari University of Venice, Italy
Feliks Hibraj	European Centre for Living Technology, Ca'Foscari University of Venice, Italy

Contents

Physical–Chemical Phenomena

Quantum Neural Networks Achieving Quantum Algorithms	3
<i>Delphine Nicolay and Timoteo Carletti</i>	
Signal Transduction and Communication Through Model Membranes in Networks of Coupled Chemical Oscillators	16
<i>Federico Rossi, Kristian Torbensen, Sandra Ristori, and Ali Abou-Hassan</i>	
Controlling Chemical Chaos in the Belousov-Zhabotinsky Oscillator	32
<i>Marcello A. Budroni, Mauro Rustici, Nadia Marchettini, and Federico Rossi</i>	
Fragment Based Molecular Dynamics for Drug Design	49
<i>Lucia Sessa, Luigi Di Biasi, Simona Concilio, and Stefano Piotto</i>	
Stochastic Numerical Models of Oscillatory Phenomena.	59
<i>Raffaele D’Ambrosio, Martina Moccaldi, Beatrice Paternoster, and Federico Rossi</i>	

Biological Systems

Understanding Embodied Cognition by Building Models of Minimal Life: Preparatory Steps and a Preliminary Autopoietic Framework	73
<i>Luisa Damiano and Pasquale Stano</i>	
Computing Hierarchical Transition Graphs of Asynchronous Genetic Regulatory Networks	88
<i>Marco Pedicini, Maria Concetta Palumbo, and Filippo Castiglione</i>	
The Impact of Self-loops in Random Boolean Network Dynamics: A Simulation Analysis	104
<i>Sara Montagna, Michele Braccini, and Andrea Roli</i>	
A Comparison Between Threshold Ergodic Sets and Stochastic Simulation of Boolean Networks for Modelling Cell Differentiation	116
<i>Michele Braccini, Andrea Roli, Marco Villani, and Roberto Serra</i>	
A Relevance Index Method to Infer Global Properties of Biological Networks	129
<i>Marco Villani, Laura Sani, Michele Amoretti, Emilio Vicari, Riccardo Pecori, Monica Mordonini, Stefano Cagnoni, and Roberto Serra</i>	

Dynamical Properties of a Gene-Protein Model.	142
<i>Davide Sapienza, Marco Villani, and Roberto Serra</i>	
Simulating Populations of Protocells with Uneven Division	153
<i>Martina Musa, Marco Villani, and Roberto Serra</i>	
An Integrated Model Quantitatively Describing Metabolism, Growth and Cell Cycle in Budding Yeast	165
<i>Pasquale Palumbo, Marco Vanoni, Federico Papa, Stefano Busti, Meike Wortel, Bas Teusink, and Lilia Alberghina</i>	
Estimating Effects of Extrinsic Noise on Model Genes and Circuits with Empirically Validated Kinetics.	181
<i>Samuel M. D. Oliveira, Mohamed N. M. Bahrudeen, Sofia Startceva, and Andre S. Ribeiro</i>	
Economy and Society	
Calibrating Dynamic Factor Models with Genetic Algorithms	197
<i>Fabio Della Marra</i>	
Functional Interactions in Complex Networks: A Three-Step Methodology for the Implementation of the Relevance Index (RI)	212
<i>Riccardo Righi and Sofia Samoili</i>	
Modeling Urbanization Perception: Emerging Topics on Hangzhou Future Sci-Tech City Development	229
<i>Debora Slanzi, Valentina Anzoise, and Irene Poli</i>	
Complexity	
Complexity Measures in Automatic Design of Robot Swarms: An Exploratory Study	243
<i>Andrea Roli, Antoine Ligot, and Mauro Birattari</i>	
Identification of “Die Hard” Nodes in Complex Networks: A Resilience Approach	257
<i>Angela Lombardi, Sabina Tangaro, Roberto Bellotti, Angelo Cardellicchio, and Cataldo Guaragnella</i>	
Optimization	
Automatic Algebraic Evolutionary Algorithms	271
<i>Marco Baiocchi, Alfredo Milani, and Valentino Santucci</i>	

Multi-objective Optimization in High-Dimensional Molecular Systems. 284
*Debora Slanzi, Valentina Mameli, Marina Khoroshiltseva,
and Irene Poli*

Multiple Network Motif Clustering with Genetic Algorithms 296
Clara Pizzuti and Annalisa Socievole

Searching Relevant Variable Subsets in Complex Systems
Using K-Means PSO 308
*Gianluigi Silvestri, Laura Sani, Michele Amoretti, Riccardo Pecori,
Emilio Vicari, Monica Mordonini, and Stefano Cagnoni*

Author Index 323

Physical–Chemical Phenomena



Quantum Neural Networks Achieving Quantum Algorithms

Delphine Nicolay^(✉) and Timoteo Carletti

Department of Mathematics, Namur Institute for Complex Systems (naXys),
University of Namur, Namur, Belgium
delphine.nicolay@unamur.be

Abstract. This paper explores the possibility to construct quantum algorithms by means of neural networks endowed with quantum gates evolved to achieve prescribed goals. First tentatives are performed on the well known Deutsch and Deutsch-Jozsa problems. Results are promising as solutions are detected for different sizes and initializations of the problems using a standard evolutionary learning process. This approach is then used to design quantum operators by combining simple quantum operators belonging to a predefined set.

1 Introduction

Quantum computation has generated a lively interest for the last two decades, since the discovery of a quantum algorithm able to factorize large integers in polynomial time [11]. In fact, the demand for better performance of computers strongly increases and quantum computation could be the answer to overcome the limitations of current computing. However, even in the case of relatively simple problems, the search for a quantum algorithm is not trivial. This fact is clearly illustrated by the parcelled development of solutions for the well known problems of Deutsch [3] and Deutsch-Jozsa [5]. Another complication of quantum computing is its physical feasibility. Indeed, quantum computing requires the development of quantum operators working on systems of qubits. Until now, researchers have been able to physically produce operators dealing with small systems composed of one or two qubits. Fortunately, it has been proved that any quantum operator can be built as a combination of these concretely realizable operators. But, once more, the development of the right combination is not a trivial problem.

In this work, we study the possibility to make use of networks endowed with quantum gates to develop appropriate quantum algorithms, i.e. appropriate combinations of quantum operators to achieve defined tasks or computations. As the construction and the learning process of these networks are roughly inspired by standard artificial neural networks, we decided to name them quantum neural networks (QNN). They are designed for their specific goals by evolutionary optimization methods. The already mentioned Deutsch and Deutsch-Jozsa problems

have been the first tasks considered for this study. We show that our methodology has led to promising results, as solutions have been detected for different sizes and initializations of the problems. Then, we have identified a set of universal quantum operators and we have applied our method to the design of quantum gates by combining operators from this set. This second phase of the research highlights an important limitation of our model which is the exponential increase of the possible combinations.

The paper is organized as follows. In Sect. 2, we remind the basic concepts of quantum computing and we present our Quantum Neural Network model. In Sect. 3, we detail the problems of Deutsch and Deutsch-Jozsa and results we get with our model. We also perform a critical discussion about our optimization methods. Section 4 presents our attempt of gates development with a set of universal quantum operators. Section 5 concludes the contribution with a summary of our results and perspectives for future work.

2 Background to Quantum Computing

2.1 Quantum Bits

The bit is the fundamental unit of classical computation. Quantum computation is developed upon a similar concept, the quantum bit, also called qubit. These qubits have basic states $|0\rangle$ and $|1\rangle$, which correspond to logical states 0 and 1 for classical bits. But, contrary to the latter ones, qubits can also be in a superposition of states

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where α and β are complex numbers constrained by the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. Usually, a qubit is considered as a vector in \mathbb{C}^2 and the basic states are then seen as a pair of orthonormal basis vector

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

As qubits are quantum objects, this superposition of states is not observable. Once the qubit is measured, the superposition is lost and the system will be found in the state $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$.

In the same way, we can define systems with n -qubit as

$$|x_n x_{n-1} \dots x_1\rangle \text{ where } x_i \in \{0, 1\} \text{ for } i = 1, \dots, n.$$

Such states can be written as a tensor product of qubits but quantum computation is much richer. Indeed, thanks to the superposition, a 2-qubit can be in the state

$$\alpha|00\rangle + \beta|11\rangle$$

which can not be constructed using tensor products of qubits. This property of quantum system is called the entanglement [9] and is proper to quantum systems.

2.2 Quantum Gates

Quantum gates, working on a qubit or an n -qubit system, are obtained using unitary operators, hence they are reversible and they respect the normalization condition. They are the basic building blocks, combined to form quantum circuits. Widely used qubit operators and their matrix representation are presented below.

- Identity operator I :

$$\begin{aligned} I|0\rangle &= |0\rangle \\ I|1\rangle &= |1\rangle \end{aligned} \quad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

- NOT operator X :

$$\begin{aligned} X|0\rangle &= |1\rangle \\ X|1\rangle &= |0\rangle \end{aligned} \quad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

- Operator Y :

$$\begin{aligned} Y|0\rangle &= i|1\rangle \\ Y|1\rangle &= -i|0\rangle \end{aligned} \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

- Operator Z :

$$\begin{aligned} Z|0\rangle &= |0\rangle \\ Z|1\rangle &= -|1\rangle \end{aligned} \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

- Hadamard transformation H :

$$\begin{aligned} H|0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ H|1\rangle &= \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \end{aligned} \quad H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

- Phase operator S :

$$\begin{aligned} S|0\rangle &= |0\rangle \\ S|1\rangle &= i|1\rangle \end{aligned} \quad S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

- $\pi/8$ operator T :

$$\begin{aligned} T|0\rangle &= |0\rangle \\ T|1\rangle &= \frac{\sqrt{2}}{2}(1+i)|1\rangle \end{aligned} \quad T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$$

The most used 2-qubit operator is the controlled-not operator (C_{not}), also called the 2-qubit XOR gate, which is represented by

$$\begin{aligned} C_{not}|00\rangle &= |00\rangle \\ C_{not}|01\rangle &= |01\rangle \\ C_{not}|10\rangle &= |11\rangle \\ C_{not}|11\rangle &= |10\rangle \end{aligned} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Its effect consists in changing the state of the second qubit if and only if the first one is equal to $|1\rangle$. In the same way, we can define other controlled gates by combining this rule and the qubits presented previously. It has been proved [1] that the controlled-not gate combined with all qubit gates form a universal set for quantum computation.

2.3 QNN Model

Our model of quantum neural networks is based on the model proposed by Deutsch [4]. The idea is to build a network whose nodes are quantum gates and connections bring quantum information through qubits. The network is obviously feedforward and the number of nodes is constant in every layer. Quantum neural networks are trained by means of heuristic optimization methods.

3 Deutsch and Deutsch-Jozsa Algorithms

3.1 Problems Description

The Deutsch [3] and the Deutsch-Jozsa [5] problems are basic problems in quantum computing. The Deutsch problem consists in deciding if a binary function $f : \{0, 1\} \rightarrow \{0, 1\}$ is constant using only one function evaluation. It is clear that this is not possible in the classical framework, where two function evaluations are needed. To achieve this goal, we have a quantum black box, called oracle, at our disposal. This oracle computes one of the four possible functions, i.e. forming all the possible couples $f(u) = v$ with $u, v \in \{0, 1\}$, by applying an unitary operator U_f defined as

$$U_f(|x\rangle|y\rangle) = |x\rangle|y \oplus f(x)\rangle$$

where $|x\rangle$ and $|y\rangle$ are the qubits of the system. The quantum circuit representing the solution of this problem is presented in Fig. 1. The sequence of operations described in this figure leads to the final state $|\psi\rangle$:

$$|\psi\rangle = \begin{cases} \pm|0\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right] & \text{if } f(0) = f(1) \\ \pm|1\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right] & \text{if } f(0) \neq f(1) \end{cases}$$

A measure of the first qubit is then sufficient to evaluate if the function is constant ($|0\rangle$) or not ($|1\rangle$).

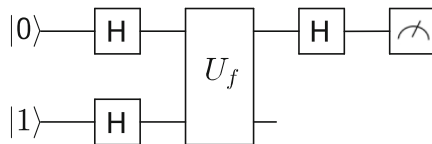


Fig. 1. Quantum circuit for the resolution of the Deutsch problem. The first qubit is initialized to $|0\rangle$ while the second one is set to $|1\rangle$. Then, an Hadamard gate is applied to the two inputs before calling the oracle. An Hadamard gate is finally applied on the first qubit, which is then measured. If it is found in the state $|0\rangle$ then the function is constant, otherwise, namely if the measure determines that the qubit is in the state $|1\rangle$, the function is not constant.

The Deutsch-Jozsa problem is a generalization of the Deutsch problem for a binary function $f : \{0, 1\}^n \rightarrow \{0, 1\}$. In this case, we have to decide if the function is constant or balanced, which means that we get 0 for half of the function evaluations and 1 for the other half. The resolution is very similar to the previous one and is presented in Fig. 2. Indeed, the qubits are initialized similarly i.e. $|0\rangle$ for the n first qubits and $|1\rangle$ for the last one. Then, an Hadamard gate is applied on all qubits before the oracle intervention. An Hadamard gate operates again on each of the n first qubits. The function is constant if all of them are finally in the state $|0\rangle$.

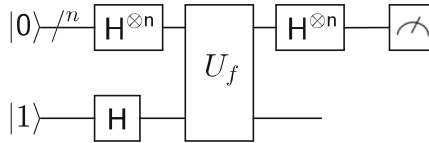


Fig. 2. Quantum circuit for the resolution of the Deutsch-Jozsa problem. The n first qubits are initialized to $|0\rangle$ while the last one is set to $|1\rangle$. Then, an Hadamard gate is applied to all qubits before calling the oracle. An Hadamard gate is finally applied on the n first qubits, which are then measured. If they are all found in the state $|0\rangle$ then the function is constant, otherwise, namely if at least one of the qubits is in the state $|1\rangle$, the function is balanced.

Even if these two problems are relatively simple, let us remark that finding their solution is not trivial. Indeed, the algorithm originally proposed by Deutsch [3] was probabilistic. It was successful with a probability of one half. In [5], Deutsch and Jozsa developed a deterministic algorithm but it required two oracle calls to succeed. The current solution, with only one function evaluation, has been proposed by Cleve et al. [2]. This shows that even in relatively simple cases, there is a need for a general strategy allowing to construct the algorithm associated to the problem at hand.

3.2 Experimentation and Results

For the trial problems of Deutsch and Deutsch-Jozsa, we have not considered a set of universal gates. The nodes could only be assigned to one of the three qubit gates I , X and H or to the oracle. Let us remind that this oracle is only used in one layer of the network, but has an effect on all qubits of the layer. Indeed, our $n + 1$ qubits, handled separately, have to be turned into a $(n + 1)$ -qubit system used as a whole by the oracle. This transformation is carried out using the Kronecker tensor product. The inverse operation is then executed after passing the oracle to recover our $n + 1$ qubits.

Quantum neural networks are evolved to solve the considered problem by a genetic algorithm (GA) [6]. The training environment contains the functions

to classify. The fitness of each individual is defined by the fraction of correct classifications. As the optimization is heuristic, all experiments have been replicated 10 times. The results presented are means on these 10 simulations¹.

The first tests on Deutsch problem have been performed with an initialization of the first qubit to $|0\rangle$ and the second one to $|1\rangle$. All simulations led to a correct solution. The only difference observed among these different solutions concerns the operator applied on the second qubit in the last layer, as it is shown in Fig. 3. This difference is not important as only the first qubit is measured to answer the asked question. This solution was already found at the first generation of the GA, this fact could be explained by the small number of possible networks (247).

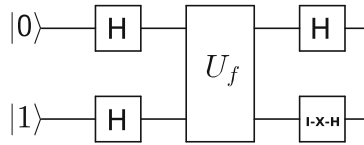


Fig. 3. Quantum circuits for the resolution of the Deutsch problem obtained with our model. The only difference among solutions pertains to the last operator applied on the second qubit, and so has no influence on the state which is measured.

Then, different parameters have been altered to observe the consequences on the learning and the final algorithm. These parameters are the number of layers in the network, the initialization of the qubits and the state to measure to be constant or balanced. When the number of layers is increased, we observe that a solution is always found even if the number of possible networks increase exponentially. Indeed, the number of admissible solutions also increase exponentially according to the number of layers. For example, if we consider five layers in the network, the two networks presented in Fig. 4 have the same effect on the quantum states.

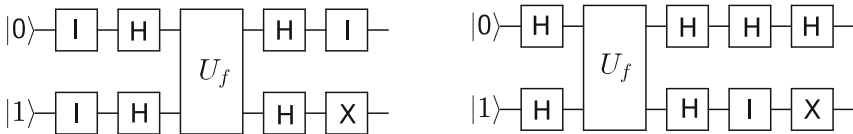


Fig. 4. Two different solutions for the problem of Deutsch if the network is formed by five layers. The networks are different but their effect on quantum bits are equivalent.

¹ The selection is performed by a roulette wheel selection. The genetic operators are the 1-point crossover and the uniform mutation. Their respective rates are 0.9 and 0.01. The population size is 100 and the maximum number of generations is 10000. The survival of best individuals is ensured by elitism.

If we exchange the initialization of the two qubits, we have to consider a network of at least four layers to find a solution. And, most of the time, the solution consists in replacing the network in the previous initialization, which means that a NOT operator is applied to each qubit in the first layer. Results are similar if we alter the initialization by setting both qubits to $|0\rangle$ or $|1\rangle$.

In case we switch the states to measure to have a constant ($|1\rangle$) or balanced ($|0\rangle$) function, we can find a solution whatever we take as initialization of our qubits. The smallest network, given in Fig. 5, is obtained if both qubits are initialized to $|1\rangle$. In other cases, the solution is made of four layers. We have also tried to look for a solution if we measure the second qubit instead of the first one but it has not worked whatever the considered initialization and configuration. This result seems consistent as such a solution has never been introduced in the literature.

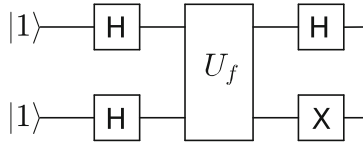


Fig. 5. Quantum circuit for the resolution of the Deutsch problem if a constant function is given by a measure of the first qubit equal to $|1\rangle$. Even if less frequently met, this scheme has already been presented in the literature [8].

Concerning the Deutsch-Jozsa problem, we have tested different sizes of the problem. Let n be the number of variables of the function, then the number of input states of the function is 2^n and the number of possible balanced functions is given by the number of combinations of 2^{n-1} units taken among 2^n . Figure 6 presents the number of possible networks according to n and the mean number of generations to reach the solution with our GA for each of this dimension. We can see in our two graphs that the increase according to n is exponential.

From $n = 3$, we have remarked that our $(n + 1)$ -qubit systems could not always be split into $n + 1$ qubits. This is due to the property of entanglement of quantum states. Indeed, some qubits that are combined with the tensor product are modified by the oracle in such a way that they can no more be separated properly. In this case, we have considered either to keep all functions or to exclude functions that lead to entangled states. In the first case, we could hardly get a fitness of 1. In the second case, we have obtained a fitness of 1 but simulations were longer as a preliminary test was needed to remove this type of functions.

3.3 Discussion on the Used Optimization Methods

Before going further, we have considered the possibility of using optimization methods different from genetic algorithms. In this way, we have implemented a simulated annealing (SA) [7] and a random search (RS). Figure 7 shows the

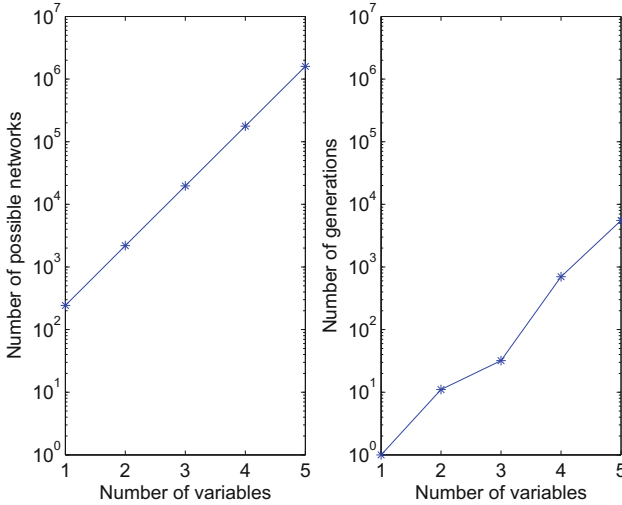


Fig. 6. Number of possible quantum networks (left panel) and number of generations to reach the solution for the Deutsch-Jozsa problem (right panel) according to the number of variables in the function.

number of iterations required by each method to reach the solution for different sizes of the Deutsch-Jozsa problem. We can observe that these numbers are very similar for the random search and the simulated annealing. Regarding our genetic algorithm, the number of required iterations is divided by a factor 100. However, this smaller number of iterations is offset by the number of function evaluations at each iteration, which is 1 for RS and SA and 100 for GA. In conclusion, the genetic algorithm and the simulated annealing do not appear more efficient than the random search.

This fact could be explained by our way of coding and modifying our model of quantum neural networks. Indeed, the shift of the oracle from one layer to another because of the application of a mutation for the GA leads to important changes in networks. This remark also holds for SA, as the oracle can be shifted during the exploration of the space of solutions. Because networks are pretty small, these big changes can modify them as strongly as it is made by random search.

Another explanation could be glimpsed by the analysis of two indicators, namely the fitness distance correlation coefficient and the autocorrelation of the function landscape [7]. As it is indicated by its name, the fitness distance correlation coefficient measures the correlation between the objective function of a candidate and its distance to the optimal solution. As for the autocorrelation, it measures the correlation between neighboring candidates. Results of these two measures for different sizes of the Deutsch-Jozsa problem are presented in Fig. 8. We can see that these two coefficients are quite low, whatever the size of the problem. This observation reinforces our intuition that GA and SA are

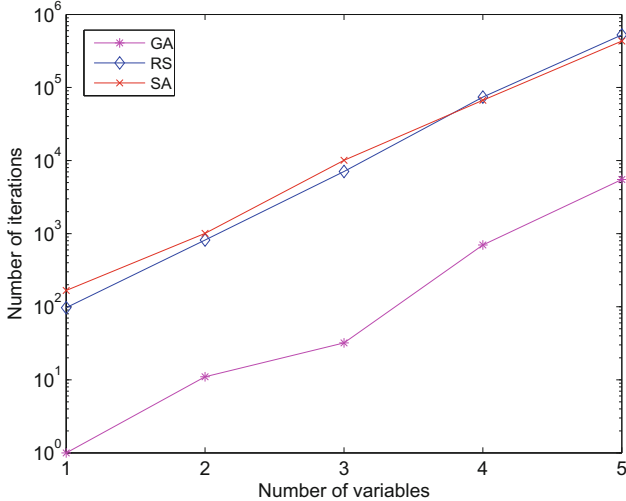


Fig. 7. Comparison of the number of iterations required by each algorithm to reach the solution. This comparison is performed for different sizes of the Deutsch-Jozsa problem. The simulated annealing and the random search required similar number of iterations while it is divided by a factor 100 for the genetic algorithm.

no more efficient than RS for this application. Indeed, if correlation does not exist between the distance to the solution and the objective function, it can not be assumed that the best individual will be found by crossovers and mutations on good individuals. Similarly, the absence of correlation between neighbors removes any advantage to an optimization method such as SA that travels from one candidate to its neighbors.

4 Quantum Gates Construction

Our methodology enables us to develop quantum algorithms solving problems of Deutsch and Deutsch-Jozsa without requiring any particular knowledge except the function to reproduce. Indeed, the appropriate algorithm appears following the learning process applied to a network composed by standard gates. Given the difficulty to develop quantum algorithms and the small number of such algorithms, we think that our results are promising even if the increase according to the number of variables is exponential. Consequently, we have considered to exploit our methodology for the implementation of quantum gates.

Our idea was to identify a set of universal gates and to develop other gates by combining those belonging to this set. We followed the statement of Nielsen and Chuang [10] and worked with a set made of 6 qubit gates to whom the controlled-not gate has been added. The qubit gates are I , H , S , T and their adjoint. As I and H are self-adjoint, we only have to add S^* and T^* .

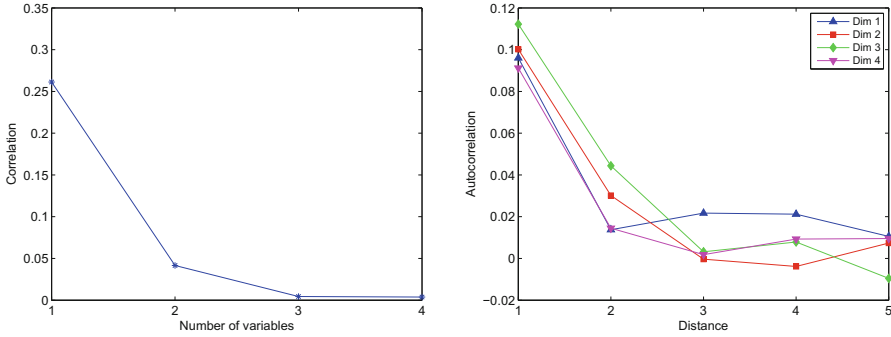


Fig. 8. Indicators analysis for different sizes of Deutsch-Jozsa problem. Left panel: fitness distance correlation coefficient. In our case, the distance between two quantum gates is fixed to 1. Moreover, we do not consider the last operator applied on the last qubit as it has any influence on the final result of the algorithm. Right panel: Autocorrelation of the objective function landscape. For this measure, we consider neighbors at distances from 1 to 5.

Before starting our optimizations, we have analyzed the two indicators presented above in order to choose the most appropriate method. For this, we have considered the objective function of the controlled-Z gate and the Toffoli gate, which is a generalization of the controlled-not for three qubits. The correlation coefficients for these two problems are respectively equal to 0.1048 and 0.2010. The autocorrelation of the function landscape is represented in Fig. 9. Once more, these measures are pretty low. Consequently, we have decided to replace our genetic algorithm by a simulated annealing. Indeed, the genetic algorithm requires more CPU time due to crossover and mutation process for analogous results. Our simulated annealing has a temperature that decreases very slowly², with the aim to explore the space of solutions as much as possible.

Firstly, we used our QNN model and our simulated annealing to design the qubit gates that were not part of the defined set, i.e. the X (NOT), Y and Z gate. The Z gate is quite easy to rebuild as it only requires a sequence of two Hadamard gates. On the contrary, X and Y respectively claim 4 and 6 layers and are represented in Fig. 10. Such a number of layers seems quite expensive for so simple gates. Then, we have succeeded in recreating the 2-qubit gates controlled- Y and controlled- Z , which are also represented in Fig. 10. Although it has been proved theoretically that all these gates could be rebuilt from a set of universal gates, let us note that we hereby provide their explicit scheme for the first time.

Nevertheless, we have quickly been confronted to one big limitation of our model, which is the exponential increase of the number of possible networks

² The temperature is initialized to 1, in such a way that a candidate decreasing the objective function by 0.5 has a probability of $\frac{2}{3}$ to be accepted. The cooling parameter is fixed to 0.99995 for a slow diminution of this probability.

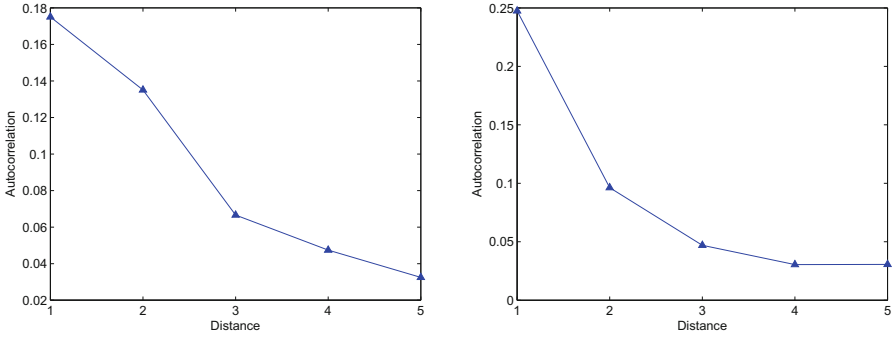


Fig. 9. Autocorrelation analysis for the objective function of two quantum gates. We consider neighbors at distances from 1 to 5. Left panel: Controlled-Z gate. Right panel: Toffoli gate.

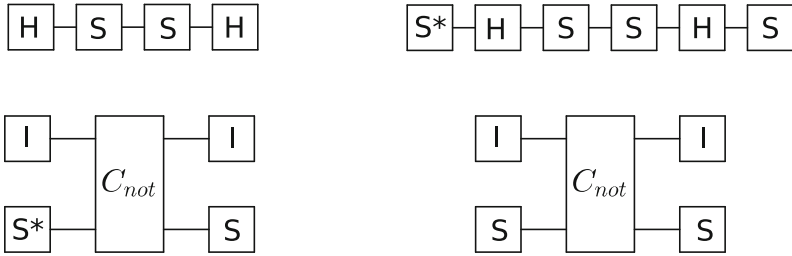


Fig. 10. Design of qubit gates with our model starting from the set of universal quantum gates. Right panel: Not (X) and controlled-Y gates. Left panel: Y and controlled-Z gates.

according to its size. Indeed, we know that a Toffoli gate requires 13 layers of three qubits to be designed from our predefined set [10]. With our model, even if we consider that we know the number of needed controlled-not gate, the number of possible networks among which the solution has to be found is superior to 10^{26} .

5 Conclusion

Quantum computation attracts considerable interest as it can be an answer to the limitations of current computers. Nevertheless, it remains difficult to elaborate quantum algorithms or quantum operators working on systems made of more than two qubits. Our aim is to study the possibility to develop a general framework based on neural networks endowed with quantum gates and evolutionary computation to tackle this difficulty.

Our approach was first used on the Deutsch and Deutsch-Jozsa problems. Results are positive as solutions were found for different configurations and different sizes of these problems. However, we have observed that our optimization method, a genetic algorithm, was no more efficient than a random search among

the space of solutions. This fact can be explained by the low values of the fitness distance correlation coefficient and the autocorrelation of the landscape, as well as by our way of coding the networks. In a second time, our QNN model has been trained to achieve quantum gates from a set of universal quantum gates.

This research highlights two limitations of our approach. The first one is linked to the entanglement property of quantum systems. Indeed, once a state is turned into an entangled state by an oracle or a controlled-not gate, we are no longer able to manage with it. The second one, and the most important for us, is the exponential increase of the networks number according to the size of this network. This increase, combined with the absence of correlation given by our indicators for the objective function, makes the resolution impossible in reasonable time for networks with more than about 15 gates.

Despite these limitations, we can envisage to improve the efficiency of our method. Firstly, we can decrease the number of possible networks by fixing the number of controlled-not gates, and stronger, by fixing the number of one qubit gates that differ from the identity. But, even with these constraints, the size of the resolvable networks will be limited. Another option would be to add a quantum operator to our set as soon as we find its breakdown. Improvements can also be imagine on the learning process. For example, we can consider the addition of a penalty in order to avoid useless sequences of operations.

Acknowledgements. We thank Andrea Roli for his critical reading of a preliminary version of the paper.

This research used computational resources of the “Plateforme Technologique de Calcul Intensif (PTCI)” located at the University of Namur, Belgium, which is supported by the F.R.S.-FNRS.

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimisation), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors.


References

1. Barenco, A., Bennett, C.H., Cleve, R., DiVincenzo, D.P., Margolus, N., Shor, P., Sleator, T., Smolin, J.A., Weinfurter, H.: Elementary gates for quantum computation. *Phys. Rev. A* **52**, 3457–3467 (1995). <https://doi.org/10.1103/PhysRevA.52.3457>
2. Cleve, R., Ekert, A., Macchiavello, C., Mosca, M.: Quantum algorithms revisited. *Proc. Roy. Soc. Lond. A: Math. Phys. Eng. Sci.* **454**(1969), 339–354 (1998). <https://doi.org/10.1098/rspa.1998.0164>
3. Deutsch, D.: Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. Roy. Soc. Lond. A: Math. Phys. Eng. Sci.* **400**(1818), 97–117 (1985). <https://doi.org/10.1098/rspa.1985.0070>
4. Deutsch, D.: Quantum computational networks. In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 425, pp. 73–90. The Royal Society (1989). <https://doi.org/10.1098/rspa.1989.0099>

5. Deutsch, D., Jozsa, R.: Rapid solution of problems by quantum computation. *Proc. Roy. Soc. Lond. A: Math. Phys. Eng. Sci.* **439**(1907), 553–558 (1992). <https://doi.org/10.1098/rspa.1992.0167>
6. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Boston (1989)
7. Hoos, H.H., Stützle, T.: *Stochastic Local Search: Foundations and Applications*. Elsevier, Amsterdam (2004)
8. Mermin, D.: *Calculs et algorithmes quantiques: méthodes et exemples*. EDP Sciences, Les Ulis (2010)
9. Mintert, F., Viviescas, C., Buchleitner, A.: Basic concepts of entangled states. In: Buchleitner, A., Viviescas, C., Tiersch, M. (eds.) *Entanglement and Decoherence*. LNP, vol. 768, pp. 61–86. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-540-88169-8_2
10. Nielsen, M.A., Chuang, I.L.: *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge (2000)
11. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **41**(2), 303–332 (1999). <https://doi.org/10.1137/S0036144598347011>



Signal Transduction and Communication Through Model Membranes in Networks of Coupled Chemical Oscillators

Federico Rossi¹ , Kristian Torbensen², Sandra Ristori³,
and Ali Abou-Hassan²

¹ Department of Chemistry and Biology “A. Zambelli”, University of Salerno,
Via Giovanni Paolo II 132, 84084 Fisciano, SA, Italy
frossi@unisa.it

<http://docenti.unisa.it/025462/en/home>

² Sorbonne Université, CNRS, PHysico-chimie des Electrolytes et Nanosystèmes
InterfaciauX, PHENIX, F-75005 Paris, France

³ Department of Chemistry and CSGI, University of Florence, Florence, Italy

Abstract. In nature, an important example of chemical communication and synchronicity can be found in cell populations where long-range chemical communication takes place over micrometer distance. *In vitro* laboratory systems can be useful to understand and control such complex biological mechanisms and, in a biomimetic approach, we present in this paper a model based on three basic features, namely (i) the compartmentalization of chemical information (using microfluidics), (ii) a stable emitter of periodic chemical signals inside compartments (Belousov-Zhabotinsky oscillating reaction) and (iii) a suitable spatio-temporal monitoring of the emitted chemical signal. In particular, starting from our recent work on the communication among oscillators *via* chemical intermediates in networks of lipid-stabilised droplets, we discuss here the role of compartments and of the geometry of the system. We present 3 different experimental configurations, namely liposomes (water-in-water dispersions), double emulsions (water-in-oil-in-water dispersions) and simple emulsions (water-in-oil dispersions) and we show that the global behaviour of networks can be influenced and controlled by several experimental parameters, like the nature of the collecting solvent, the presence of dopants and the network geometry. Numerical models supporting and explaining the experimental findings are also discussed.

Keywords: Belousov-Zhabotinsky reaction · Microfluidics
Lipid droplets · Chemical oscillators network · Chemical coupling

1 Introduction

Biological systems are the most fascinating expression of self-organisation phenomena taking place in nature. After Prigogine’s work, self-organisation is

interpreted as the tendency of far-from-equilibrium systems, also known as *Dissipative Structures*, to spontaneously organise in more complex assemblies, starting from simple elements. Such kind of organisation bears new features that are in a stationary state, kept far from thermodynamic equilibrium by a constant flux of energy and/or matter [1–3]. As an extension, several researchers considered Life and many of its manifestations as dissipative structures, providing, for instance, a possible solution to the evolutionary problem of order out of disorder in the transitional stages between abiotic and prebiotic ages [4]. Beside the high hierarchical structures, like in biological systems, there are simple physical and chemical systems that manifest self-organising properties, such as, for example, the Belousov-Zhabotinsky (BZ) chemical oscillator [5,6]. Starting from the seventies, chemical oscillators quickly became a simple model for studying complex phenomena typical of the living realm, such as oscillations, bistability, excitability and pattern formation [7,8].

However, equilibrium dynamics is also fundamental for understanding the beautiful complexity of nature. In this respect, similarly to self-organisation, self-assembly has been defined as the tendency of single components to spontaneously aggregate in complex structures while tending to a minimum (or a maximum) of a thermodynamic potential [9–11]. In chemical and biochemical fields, dispersed media like micelles, emulsions and liposomes are genuine examples of self-assembling systems [12]. Unlike to Dissipative Structures, such systems do not need a continuous flux of energy to survive [9,10,13].

In this context, blending the structural properties of self-assembled matrixes together with the evolutive peculiarities of dissipative system, allows to study an important aspect of biological systems, namely the transmission of signals across an amphiphilic boundary layer (membrane) and the synchronisation and communication among coupled chemical systems in networks [14,15]. The Epstein's group in Brandeis pioneered the study of coupled chemical oscillators in water-in-oil nano size domain (microemulsions), where the interfacial film was a simple AOT monolayer [16,17] to find that, the exchange of molecules among water compartments dispersed in a nonpolar solvent and containing the BZ reaction, produced a rich variety of structured patterns at the macroscopic level. The cooperative behaviour of nano-droplets, mediated by diffusion (or cross-diffusion [18,19]) of chemical messengers, resulted in an unexpected and emergent global behaviour at a higher hierarchical level. By using microfluidics technique, the microemulsion system was then upscaled to an emulsion system with the droplets having a characteristic size of hundreds of micrometers; here it was found that a network of diffusively coupled oscillators could produce global in-phase and out-of-phase oscillations, or more complex dynamical behaviours, depending whether the messenger molecules were activators, inhibitors or a mix of the two, respectively [20–22]. Other groups used microfluidics devices to explore similar configurations [23–25].

In a more realistic biomimetic approach, our group substituted synthetic surfactants with phospholipids to stabilise droplets in dispersed systems. We could thus study pattern formation in membrane model systems [26–28] and, by using microfluidics [29], chemical communication in liposomes [30], double

emulsions [31] and emulsions [32]. The oscillating BZ reaction was employed as the signal generator. The overall reaction is driven by the oxidation of an organic substrate, e.g. malonic acid (MA), by bromate in acidic solution in the presence of a catalytic species in the form of an organometal complex, such as ferriin [a phenanthroline-iron(II) complex]. The oscillatory dynamics, however, is governed by the amount of the inhibitory intermediate bromine and the excitatory intermediate bromous acid. These BZ intermediates might diffuse between individual microdrops, thus affecting the overall oscillatory dynamics and synchrony of multiple drop arrays. As such, the intermediates serve as messenger molecules between individual drops. More details about the kinetic mechanism responsible for oscillations will be given in Sect. 3. 1,2-dimyristoyl-*sn*-glycero-3-phosphocholine (DMPC) is known to form stable bilayers in water and to self-assemble spontaneously at the water-oil interface giving resistant, yet internally fluid, membranes [33,34]. DMPC was thus used in our experiments as the principal component of the membranes.

In this paper we present 3 different experimental configurations, namely liposomes (water-in-water dispersions), multi-core double emulsions (water-in-oil-in-water dispersions) and simple emulsions (water-in-oil dispersions), whose structures are sketched in Fig. 1, and we discuss how the global behaviour of networks can be influenced and controlled by the nature of the collecting solvent, the presence of dopants and the network geometry. Numerical models supporting and explaining the experimental findings will be also discussed.

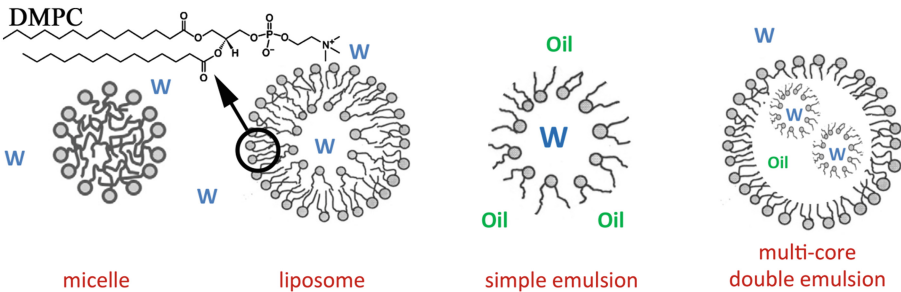


Fig. 1. Sketch of the structure of the dispersed media used in this work. The phospholipid DMPC is the amphiphilic molecule stabilizing the dispersions. W stands for water and with Oil is intended a generic nonpolar solvent

2 Experimental Approach: Microfluidic Techniques for the Generation of Oscillating Droplets

Microfluidic techniques are a reliable and easy method to synthesize droplets (either single and double emulsions, liposomes or polymersomes) with controllable size, monodispersity and composition [29]. In particular, to generate both

liposomes and emulsions loaded with the BZ reaction, we employed a home-made coaxial flow microfluidic device adapted from the setup devised by the group of D. Weitz [35]. The geometry of our setup varied depending on the experimental configuration we explored. In particular, we studied three different systems: planar 2-D liposomes and multi-core double emulsions and planar 1-D arrays of single emulsions. Figure 2 resumes the experimental conditions and the experimental observations of the most significative results in liposomes (left panel) and multi-core double emulsions (right panel). Experimental details are in the figure captions and in references [30, 31]. From a physical point of view, the two systems have different solvents separating the oscillating droplets; this fact implies the presence of an osmotic pressure in the case of liposomes that has to be balanced by adding an electrolyte in the collecting solution. Moreover, the different solubility properties of the surrounding solvents might affect the communication among the oscillators. In fact, these features are reflected in the dynamical behaviour of the two systems. In the case of liposomes we observed the development of autocatalytic fronts in single droplets, that could be transmitted from one compartment to the neighbours. In the series of pictures b–e of Fig. 2 an autocatalytic front starts to oxidise the droplet 1, wherein the colour change from dark (reduced form of the catalyst, ferroin) to bright (oxidised form of the catalyst, ferriin). The oxidative front is then propagated *via* the transduction of a chemical signal through the liposomes membrane to the surrounding droplets, as it is evident from the colour change of droplets 2, 5 and 6. The signal transmission sequence among liposomes 2, 5 and 6 was analysed by means of the space-time (ST) plot reported in Fig. 2f. Thin slices cut from each frame along the white line in Fig. 2e were vertically stacked, so that the horizontal axis represents the actual space spanned by the oxidation pulse ($\sim 820 \mu\text{m}$) and the vertical axis represents the time elapsed from the generation of the first pulse in liposome 2 to the end of the last pulse in liposome 6 ($\sim 23 \text{ s}$). The reciprocal of the slope of the diagonal borders between dark and bright areas represents the speed at which the chemical pulses travel inside the water compartment of the liposome and was calculated to be in the range $110\text{--}150 \mu\text{m s}^{-1}$. From the ST plot we could also quantify the lag time in about 5 s, during which a liposome remains in an oxidised state before transmitting the impulse to its neighbour.

In liposomes experiments, we demonstrated an actual communication among different single oscillators. The oxidative pulse transmission suggested the autocatalytic species, HBrO_2 , as the main messenger molecule able to cross the DMPC bilayers and this was confirmed by an electrochemical investigation of the membranes during the oscillatory cycles [30, 36]. However, mainly because of the osmotic pressure, the stability of liposomes was not long enough ($< 10 \text{ min}$) to study the network dynamics during a series of sustained oscillations. Therefore we devised a series of experiments in a double emulsion system as depicted in Fig. 2g and h, where the DMPC stabilised oscillating droplets were dispersed in an organic solvent (the same used for the encapsulation process) and, in turn, in a PVA (Polyvinyl Alcohol) solution. In contrast to liposomes, in double emulsions the DMPC membrane around

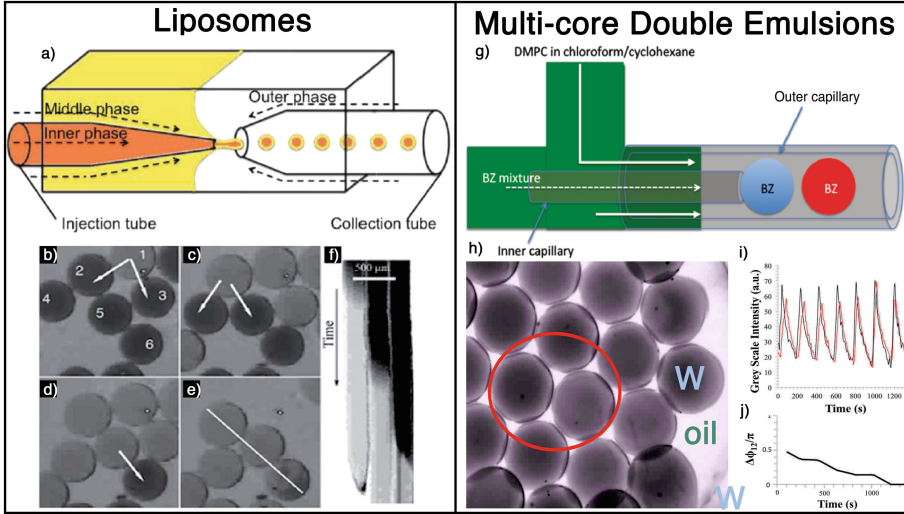


Fig. 2. Experimental setup and dynamical behaviour in liposomes (left panel) and multi-core double emulsions (right panel). (a) Microfluidic device for liposomes generation: the inner phase was a BZ aqueous mixture having H_2SO_4 (300 mM), NaBrO_3 (120 mM), MA (30 mM) and ferroin (5 mM), the middle phase contained DMPC solubilised in a mixture of chloroform: cyclohexane (40:60; v/v), the external phase was an aqueous solution of PVA (Polyvinyl alcohol) (7%, wt), an aqueous solution of NaBrO_3 (0.4 M) was used to recuperate the liposomes at the exit of the microfluidic device; (b)–(e) pulse transmission across the touching liposomes after solvent evaporation. White arrows indicate the direction of pulse propagation; (f) space-time plot of liposomes 2, 5, 6 along the white bar in panel (e); (g) Microfluidic device for double emulsion generation. Conditions are the same as in (a) but in this case the droplets of the surrounding oil phase where collected in a PVA solution only at the end of the encapsulation process, preventing the evaporation of the organic solvents, H_2SO_4 (350 mM), NaBrO_3 (180 mM), MA (150 mM) and ferroin (2.5 mM); (h) Final configuration of the oscillating droplets in double emulsions; (i) Time series of the oscillating dynamics of the two droplets in the red circle in (h); (j) Temporal behaviour of the phase difference between the two droplets calculated by using the Eq. (1). (Color figure online)

each droplet is a thick layer (30–50 nm) with disordered internal structure. By avoiding the osmotic pressure problems, we could obtain droplets stable enough to record more than 10 oscillatory cycles (~ 30 min) and observe an in-phase synchronisation tendency between touching droplets. Figure 2i shows the timeseries extracted from the two droplets in the red circle of Fig. 2h; it is quite evident that, after few oscillations, the two droplets tend to spike with the same period and phase. This is also confirmed by the evolution of the phase difference between the two oscillators ($\phi_{12} = \phi_1 - \phi_2$) reported in Fig. 2j and calculated by means of the Eq. (1) [37,38].

$$\phi_i(t) = 2\pi \frac{t - t_k}{t_{k+1} - t_k} + 2k\pi \quad t_k < t < t_{k+1} \quad (1)$$

where t_k is the time of the k -th peak of the oscillatory time series of the oscillator i . The diagonal lines in Fig. 2j mean that the phase difference is changing in time, while horizontal lines indicate that both oscillators have the same period and oscillate with a constant phase difference. This represents a *phase lock* state, *i.e.* a coherent behaviour of the two oscillators. The phase difference, in particular, shifts from an initial value of about 0.5 p during the first cycles to 0 at the end, indicating that the two oscillators adjust their oscillation frequency until they reach a synchronistic behaviour. The in-phase oscillations of the communicating droplets reveals an activatory coupling path, that can be brought about by the exchange of the autocatalytic intermediates [39], in agreement with the electrochemical investigations and with the pulse transmission in liposomes. Simulations presented in the Sect. 3 confirmed this hypothesis.

1-D arrays of oscillating droplets were built to explore a linear connection geometry in a controlled and tunable configuration. By taking advantage of the microfluidic setup shown in Fig. 3a, it was possible to obtain a reliable and robust network of oscillators that could be monitored for longer periods with respect to double emulsions. In this case, we dealt with a single emulsion system obtained by keeping the droplets inside of the collection tube (Fig. 3b). Experimental

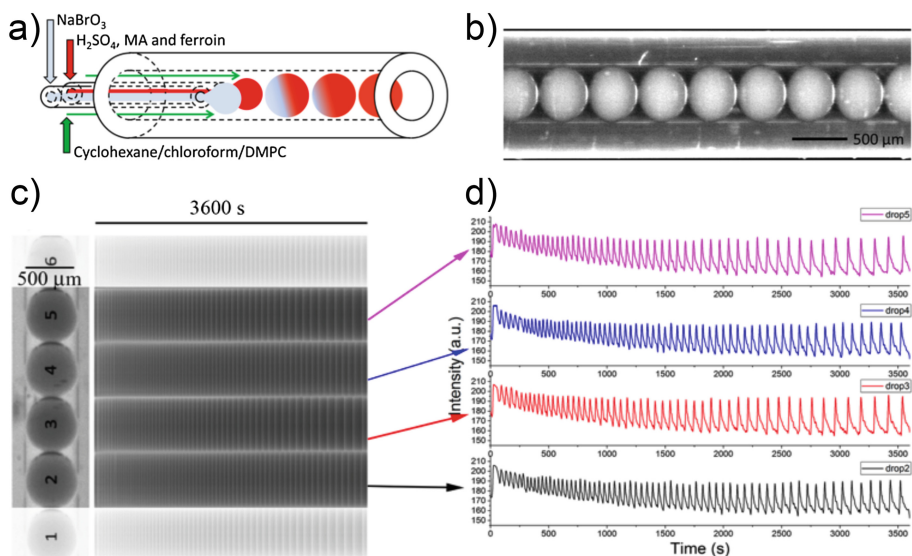


Fig. 3. (a) Sketch of the microfluidic device used to generate the droplet arrays; (b) 1D array of BZ containing droplets collected in a PTFE tube for monitoring. (c) Array of six oscillating droplets in the simple emulsion system. Space-Time plots of each droplet were reconstructed from the movie frames (sampling time 1 s). (d) Time-series extracted from the Space-Time plots by converting the pixels in grey scale values; H_2SO_4 (300 mM), NaBrO_3 (120 mM), MA (300 mM) and ferroin (5 mM), The suspending oil phase consisted of a mixture of chloroform/cyclohexane (1:2), DMPC (0.8% w/w), and STS (0.2% w/w).

details are in the figure caption and in reference [32]. In order to work at a low lamellarity of the membranes between the touching droplets, thus facilitating the exchange of messenger molecules, we used sodium tetradecyl sulfate (STS), since charged surfactants are known to favour oligo- (or mono-) lamellar structures in liposomes, to dope the DMPC [32, 40]; cholesterol (CHOL) was also used to interact with brominated BZ intermediates and tune the communication between droplets [32, 40].

As an example of the operative setup, Fig. 3c shows a simple emulsion array of six oscillating droplets loaded with BZ and surrounded by the mixture of cyclohexane/chloroform containing DMPC+STS. On the right of the array, the ST plots display the oscillating dynamics of each droplet. The vertical bright lines correspond to a firing of the oscillator (oxidized catalyst), whilst the dark regions represent the recovery period (reduced catalyst) between single oscillations. From the ST plots of four droplets (2–5), the time-series were extracted; the corresponding time-series are reported in Fig. 3d.

The analysis of the timeseries for the DMPC+STS system revealed an anti-phase global dynamics among adjacent droplets ($\Delta\phi \sim \pi$) and an in-phase synchronisation among alternating droplets ($\Delta\phi \sim 0$), as highlighted in Fig. 4a–b. This behaviour is typical of networks of oscillators coupled *via* inhibitory signals, that in the BZ case are generally Br_2 or Br^- intermediates. To confirm the prominent role of brominated species as inhibitory messengers in the simple emulsions system, we modified the membrane composition by inserting CHOL molecules in order to modulate the communication pathway and to seek a difference in the global behaviour of the array. We expected, in fact, that cholesterol-doped DMPC membranes act as a barrier for Br_2 , thus preventing, or at least mitigating, inhibitory coupling. The analysis of the phase difference of the droplets couples reported in Fig. 4c–d, shows a substantially erratic behaviour for all the permuted couples except for a weak coupling of the droplets 3,4.

In this section, we showed that the global behaviour of a network of coupled oscillators can be influenced and controlled by several experimental parameters: (i) the nature of the collecting solvent determines the type of dispersion (liposomes, double and single emulsions) and control the osmotic pressure; (ii) the presence of dopants like STS favour communication among droplets by decreasing the lamellarity of the membranes; (iii) CHOL in the membranes selectively interacts with messenger molecules; (iv) finally, also the network geometry exerts a certain influence on the type of communication pathways among individual oscillators.

In the next section we propose few models that can explain and reproduce some of the observed results.

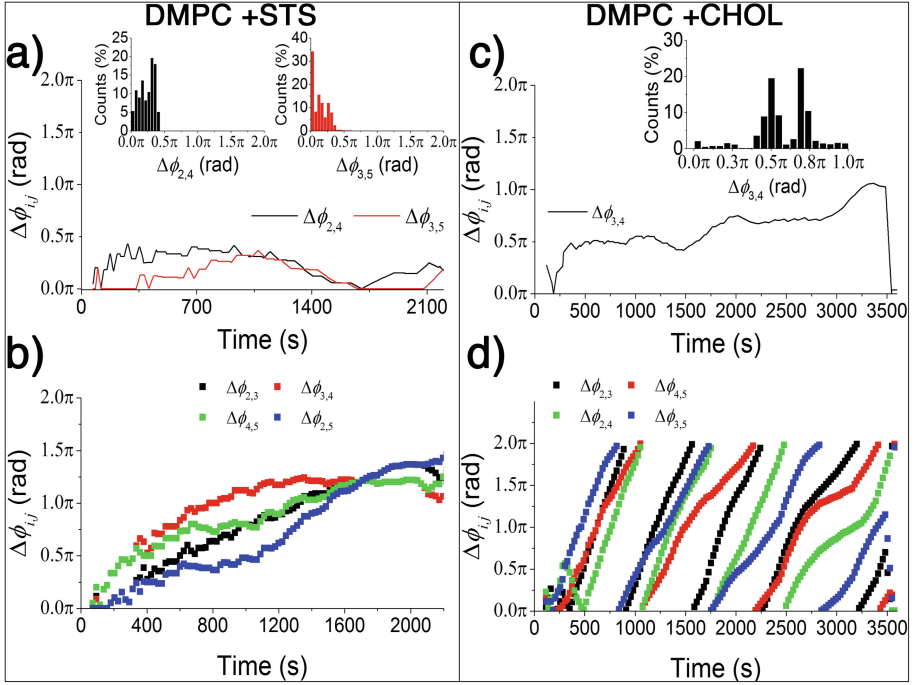


Fig. 4. (a) Phase difference of the alternate droplets 2,4 and 3,5, as a function of time. The inset shows the count distribution for various intervals of phase differences in the DMPC+STS system; (b) Phase differences of adjacent droplet couples vs time, displaying the progressive phase shift to anti-phase oscillations in the DMPC+STS system; (c) Phase difference of the droplets 3,4 as a function of time, the inset shows the count distribution for various intervals of phase differences in the DMPC+CHOL system; (d) Phase differences of the droplet couples 2,3; 2,4; 4,5; and 3,5 vs time, displaying an uncorrelated coupling for these adjacent and alternate droplets in the DMPC+CHOL system.

3 Modeling

The first step for understanding the dynamics of the droplets network is to model the chemical system responsible for the oscillations. A minimal reaction mechanism that reproduces the complex chemistry of the ferroin catalysed BZ reaction is represented by the reaction scheme (R1)–(R14) with relative kinetic constants reported in Table 1 [32, 41–43]. According to the classic interpretation of the oscillatory mechanism [44], the reactions (R1)–(R14) can be simplified in three main processes as sketched in the lower panel of Fig. 5. Process A accounts for the reactions (R1)–(R4) and it is dominated by the inhibitor Br^- chemistry, process B accounts for the reactions (R5)–(R8) and represents the autocatalytic reactions that involve the activator HBrO_2 , process C accounts for reactions (R9)–(R14) and it is responsible for the regeneration of the catalyst ($\text{M}^{(\text{ox})} \rightarrow \text{M}^{(\text{red})}$) and the

production of the inhibitor that, in turn, restarts the cycle. A kinetic scheme can then be derived from the three processes and numerically integrated for simulating the dynamics of the systems (legend for the symbols is reported in the Fig. 5 caption).

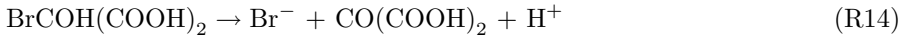
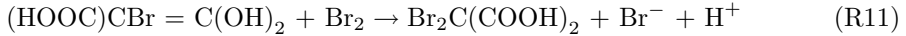
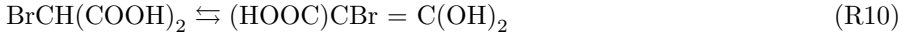
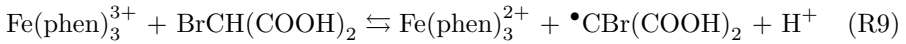
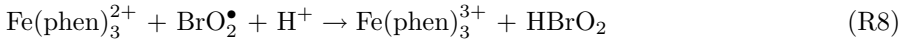
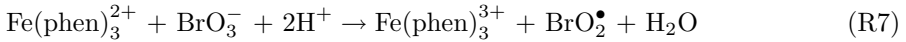
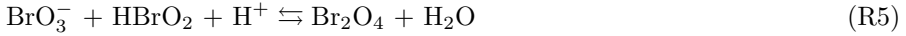
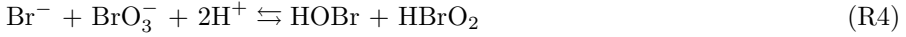


Table 1. Forward and backward reaction rates for the BZ model involving reactions (R1)–(R14). Taken from refs. [32,41–43].

Reaction	k_{forward}	k_{inverse}
R1	$8 \times 10^9 \text{ mol}^{-2} \text{ dm}^6 \text{ s}^{-1}$	80 s^{-1}
R2	$2.5 \times 10^6 \text{ mol}^{-2} \text{ dm}^6 \text{ s}^{-1}$	
R3	$3 \times 10^3 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$	
R4	$10 \text{ mol}^{-3} \text{ dm}^9 \text{ s}^{-1}$	$3.2 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$
R5	$48 \text{ mol}^{-2} \text{ dm}^6 \text{ s}^{-1}$	$3.2 \times 10^3 \text{ s}^{-1}$
R6	$7.5 \times 10^4 \text{ s}^{-1}$	$1.4 \times 10^9 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$
R7	$0.38 \text{ mol}^{-3} \text{ dm}^9 \text{ s}^{-1}$	
R8	$1 \times 10^9 \text{ mol}^{-2} \text{ dm}^6 \text{ s}^{-1}$	
R9	$100 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$	$6 \times 10^8 \text{ M}^{-2} \text{ s}^{-1}$
R10	0.012 s^{-1}	800 s^{-1}
R11	$3.5 \times 10^6 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$	
R12	$6.6 \times 10^4 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$	
R13	$1 \times 10^8 \text{ mol}^{-1} \text{ dm}^3 \text{ s}^{-1}$	
R14	1.5 s^{-1}	

When communication among droplets is considered, the system can be modelled as sketched in Fig. 5a for liposomes and double emulsions and b for the droplets array: each droplet contains the BZ reaction and it is free to exchange the activator, HBrO_2 , and the inhibitor, Br_2 , with neighbours through a simple equilibrium reaction following mass action kinetics. The coupling reactions are reported in the lowest part of Fig. 5 with the corresponding kinetics to be included in the BZ scheme. Details are reported in the figure caption. DMPC was

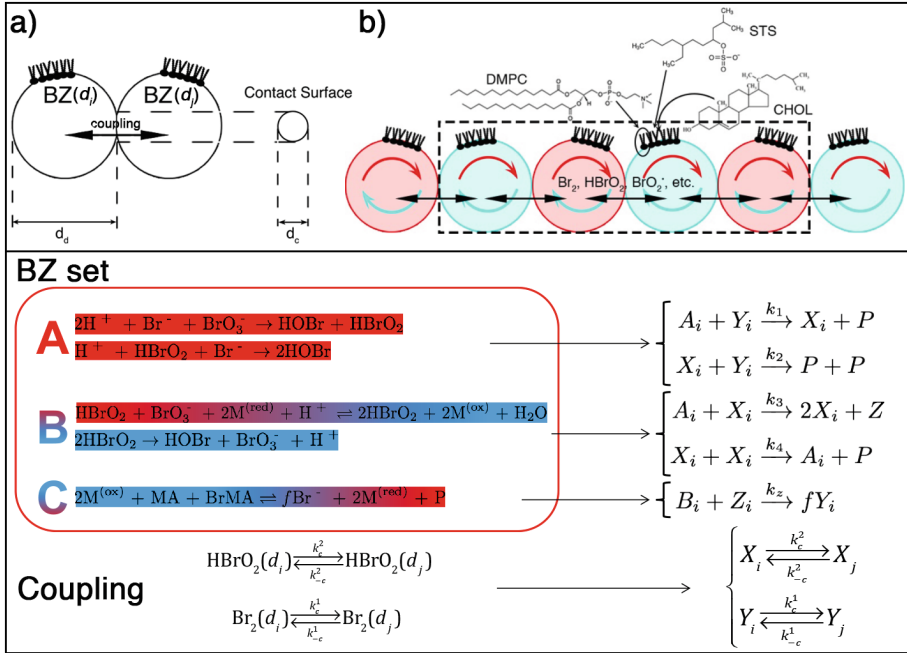


Fig. 5. (a) Sketch of two touching droplets. (d_i) represents the droplet, d_d is a droplet diameter and d_c is the diameter of the contact surface between two droplets, which has been approximated to a circle; (b) Model of the BZ oscillator array.

The lower part of the panel shows the most important processes accounting for the chemistry of the BZ reaction: X_i , Y_i and Z_i represent the concentration of HBrO_2 , Br^- and M^{ox} in droplet i , respectively; the kinetic constants derived from Table 1 are k_1 ($\text{M}^{-1}\text{s}^{-1}$) = 0.245, k_2 ($\text{M}^{-1}\text{s}^{-1}$) = 1.05×10^6 , k_3 ($\text{M}^{-1}\text{s}^{-1}$) = 14.7, k_4 ($\text{M}^{-1}\text{s}^{-1}$) = 1.05×10^3 , k_z ($\text{M}^{-1}\text{s}^{-1}$) = 1, $f = 0.5$ is a stoichiometric factor which accounts for the Br^- regeneration. $k_c^i = k_{-c}^i$ (s^{-1}) are the transfer kinetic constants related with the permeability of the i -th species, P_m^i (cm/s), towards the phospholipid membranes by the relation $k_c^i = P_m^i A_c / V_d$, where V_d is the droplet volume and A_c is the contact surface area between two droplets. The values for V_d and A_c were determined from experiments, the value for P_m^1 is 0.07 cm/s [45], the value for P_m^2 was chosen as 1×10^{-4} cm/s [46], so that $k_c^1 = 0.15 \text{ s}^{-1}$ and $k_c^2 = 2 \times 10^{-4} \text{ s}^{-1}$.

Colours map the state and the transition of the catalyst forms, red for the reduced and blue for the oxidised state. (Color figure online)

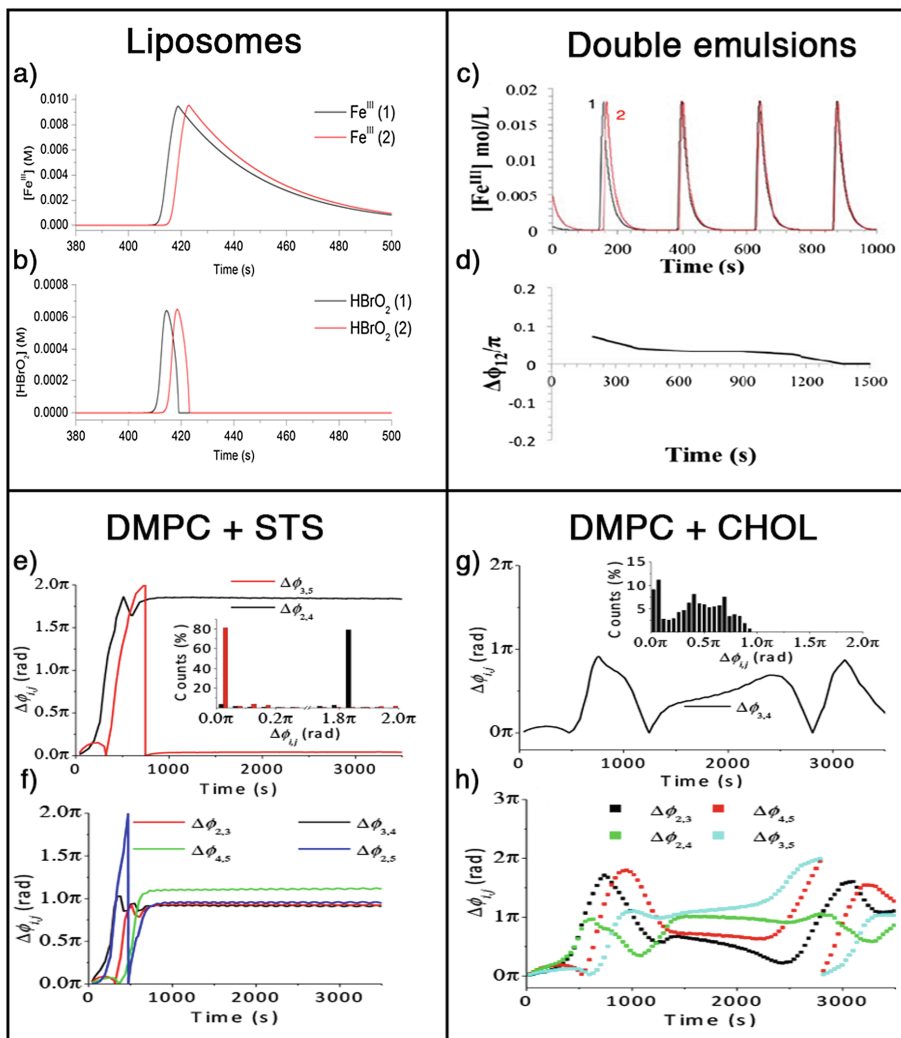
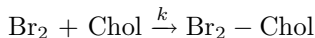


Fig. 6. (a)–(b) Signal transmission between two liposomes. At $t = 400$ s a signal has been triggered in droplet 1 ($[\text{HBrO}_2]_{\text{ex}} = 1.1 \times 10^{-6}$ M). After about 4 s the signal reaches droplet 2 and causes the production of the autocatalytic species and the consequent oxidation of the ferrous; (c) Numerical simulations of the coupled dynamics of two BZ droplets in a double emulsion system. (d) Time evolution of the phase difference calculated from the time series in the panel (c); (e)–(h) Simulated phase dynamics for four droplets in an array of linearly coupled oscillating simple emulsions: (e) Phase difference for the synchronised droplets in the DMPC+STS system; (f) Phase difference for the adjacent droplets oscillating in anti-phase in the DMPC+STS system; (g) phase difference for the weakly coupled droplets (3,4) in the DMPC+CHOL system; (h) phase difference for adjacent and alternate droplets showing uncorrelated phase behaviour over time.

always used as the major structural unit for the mono- and bilayer membranes, the latter present at all the droplet interfaces, alternatively intercalated with CHOL and STS molecules in the case of arrays.

When cholesterol was intercalated in the DMPC membranes, a fast bromination reaction was added to the scheme, similarly to the bulk systems we investigated in reference [40]



with a kinetic rate constant $k = 340 \text{ M}^{-1}\text{s}^{-1}$.

Numerical simulations were performed by integrating the kinetic scheme by means of the CO.PA.SI. software [47]. All data used in simulations reflected the real experimental parameters, and the different initial conditions of the droplets were reproduced by introducing a small delta in the concentration of the reactants ($\pm 0.1\%$). All the simulations details can be found in references [30–32, 40], Fig. 6 resumes the most important numerical results that reproduce the experimental findings presented in Sect. 2.

The pulse transmission between two communicating liposomes (Fig. 6a and b) was clearly reproduced on a timescale comparable with the experiments ($\sim 4 \text{ s}$). In this case we simply perturbed one droplet in its excitable state and we followed the time evolution of the impulse in a second droplet. Both the autocatalysis and the oxidation of the catalyst take place with a lag time consistent with the experimental data reported in Fig. 2f.

Phase synchronisation of two communicating droplets in a double emulsion system was also reproduced by numerical simulations (Fig. 6c and d), both the oscillation profiles and the evolution of the phase difference are in excellent agreement with the experimental results reported in Fig. 2i and j.

Finally the typical behaviour of the arrays of simple emulsions, either in the presence of STS or CHOL as dopants in the membranes (Fig. 4), was reproduced by the model as showed by the results in Fig. 6e–h.

4 Conclusions

In this paper we discussed how the nature of the compartments, of the solvents and of the network geometry influenced the communication among chemical oscillators in networks of lipid-stabilised droplets. We employed the Belousov-Zhabotinsky reaction as the source/sink of chemical signals transmitted from and to single network elements. The chemical signals directly influenced the time evolution of each droplet, that, in turn, creates a feedback to the network. Microfluidic techniques allowed precise and reliable control over the experimental conditions, thus we could explore several network configurations in different chemical environments. By using this approach, it can be relatively simple to follow the global dynamics of large networks of far-from-equilibrium reactions, that can mimic the complex behaviours typical of the biological systems (self-organisation and self-regulation, oscillations, pattern formation etc.).

Moreover, with respect to the previous work on similar systems, the use of lipids as barrier-forming molecules confers to the overall structure an enhanced biomimetic character.

We investigated 3 different experimental configurations as a function of the environment where the droplets were dispersed, namely liposomes (water-in-water dispersions), double emulsions (water-in-oil-in-water dispersions) and simple emulsions (water-in-oil dispersions). The lipid molecule DMPC was always used as the major structural unit alternatively intercalated with STS or CHOL molecules to tune the communication properties.

We showed that the global behaviour of networks can be influenced and controlled by several experimental parameters, like the nature of the collecting solvent, the presence of dopants and the network geometry. The most important molecules responsible for communication were identified in the brominated species, being the inhibitor Br_2 and the autocatalytic activator HBrO_2 the ones chosen for numerical simulations. In liposomes and double emulsions the communication was dominated by the activators (pulse transmission and in-phase oscillations), in contrast to the 1-D arrays where the communication between adjacent droplets mainly exhibited an inhibitory character (anti-phase oscillations), governed by the prominent role of Br_2 . In the presence of mono-lamellar membranes, in fact, molecular bromine has a higher permeability with respect to the activator HBrO_2 . This is also confirmed by the experiments with bromine-blocking molecule (i.e. cholesterol) intercalated in the membrane structure; in this case, the global dynamics resulted in a weakly coupled array with an erratic global behaviour. Numerical simulations of coupled oscillators (up to 6 units) confirmed our hypothesis and could reproduce, qualitatively and quantitatively, the experimental observations.

Acknowledgments. F.R. gratefully acknowledges the University of Salerno for the grants ORSA158121 and ORSA167988. F.R. and A.A.H. acknowledge the support through the COST Action CM1304 (Emergence and Evolution of Complex Chemical Systems).

References

1. Prigogine, I.: Time, structure and fluctuations. In: Nobel Lectures, Chemistry 1971–1980, pp. 263–285. World Scientific Publishing Co., Singapore (1977)
2. Nicolis, G., Prigogine, I.: Self-organization in Nonequilibrium Systems. Wiley, New York (1977)
3. Field, R.J., Burger, M.: Oscillations and Traveling Waves in Chemical Systems. Wiley, New York (1985)
4. Ruiz-Mirazo, K., Briones, C., de la Escosura, A.: Prebiotic systems chemistry: new perspectives for the origins of life. *Chem. Rev.* **114**(1), 285–366 (2014)
5. Belousov, B.P.: A periodic reaction and its mechanism. In: *Sbornik Referatov po Radiatsionno Meditsine*, Moscow, Medgiz, pp. 145–147 (1958)
6. Zhabotinsky, A.M.: Periodic liquid phase reactions. *Proc. Acad. Sci. USSR* **157**, 392–395 (1964)

7. Winfree, A.T.: *The Geometry of Biological Time*. Springer, Heidelberg (2001). <https://doi.org/10.1007/978-3-662-22492-2>
8. Tiezzi, E.: *Steps Towards an Evolutionary Physics*. WIT Press, Southampton (2006)
9. Yamaguchi, T., Suematsu, N., Mahara, H.: Nonlinear dynamics in polymeric systems. In: Pojman, J.A., Tran-Cong-Miyata, Q. (eds.) *Nonlinear Dynamics in Polymeric Systems*. Volume 869 of ACS Symposium Series, Washington DC, pp. 16–27 (2004)
10. Yamaguchi, T., Epstein, I.R., Shimomura, M., Kunitake, T.: Introduction: engineering of self-organized nanostructures. *Chaos* **15**(4), 047501-1–047501-3 (2005)
11. Gompper, G., Domb, C., Green, M.S., Schick, M., Leibowitz, J.L.: *Phase Transitions and Critical Phenomena: Self-assembling Amphiphilic Systems*. Academic Press, Cambridge (1994)
12. Cevc, G.: *Phospholipids Handbook*. CRC Press, Boca Raton (1993)
13. Fennell-Evans, D., Wennerström, H.: *The Colloidal Domain: Where Physics, Chemistry, Biology, and Technology Meet*. Wiley, Hoboken (1999)
14. Lach, S., Yoon, S.M., Grzybowski, B.A.: Tactic, reactive, and functional droplets outside of equilibrium. *Chem. Soc. Rev.* **45**, 4766–4796 (2016)
15. Ashkenasy, G., Hermans, T.M., Otto, S., Taylor, A.F.: Systems chemistry. *Chem. Soc. Rev.* **46**(9), 2543–2554 (2017)
16. Vanag, V.K., Epstein, I.R.: Pattern formation in a tunable medium: the Belousov-Zhabotinsky reaction in an aerosol OT microemulsion. *Phys. Rev. Lett.* **87**(22), 228301–4 (2001)
17. Epstein, I.R., Xu, B.: Reaction-diffusion processes at the nano- and microscales. *Nat. Nanotechnol.* **11**(4), 312–319 (2016)
18. Rossi, F., Vanag, V.K., Epstein, I.R.: Pentanary cross-diffusion in water-in-oil microemulsions loaded with two components of the Belousov-Zhabotinsky reaction. *Chem. Eur. J.* **17**(7), 2138–2145 (2011)
19. Budroni, M.A., Lemaigre, L., De Wit, A., Rossi, F.: Cross-diffusion-induced convective patterns in microemulsion systems. *Phys. Chem. Chem. Phys.* **17**(3), 1593–1600 (2015)
20. Toiya, M., Vanag, V.K., Epstein, I.R.: Diffusively coupled chemical oscillators in a microfluidic assembly. *Angew. Chem. Int. Ed.* **47**(40), 7753–7755 (2008)
21. Delgado, J., Li, N., Leda, M., González-Ochoa, H.O., Fraden, S., Epstein, I.R.: Coupled oscillations in a 1D emulsion of Belousov-Zhabotinsky droplets. *Soft Matter* **7**(7), 3155 (2011)
22. Tompkins, N., Li, N., Girabawe, C., Heymann, M., Ermentrout, G.B., Epstein, I.R., Fraden, S.: Testing Turing’s theory of morphogenesis in chemical cells. *Proc. Natl. Acad. Sci.* **111**(12), 4397–4402 (2014)
23. Thutupalli, S., Herminghaus, S., Seemann, R.: Bilayer membranes in microfluidics: from gel emulsions to soft functional devices. *Soft Matter* **7**(4), 1312 (2011)
24. de Souza, T.P., Perez-Mercader, J.: Entrapment in giant polymersomes of an inorganic oscillatory chemical reaction and resulting chemo-mechanical coupling. *Chem. Commun.* **50**(64), 8970–8973 (2014)
25. Guzowski, J., Gizynski, K., Gorecki, J., Garstecki, P.: Microfluidic platform for reproducible self-assembly of chemically communicating droplet networks with pre-designed number and type of the communicating compartments. *Lab Chip* **16**(4), 764–772 (2016)

26. Magnani, A., Marchettini, N., Ristori, S., Rossi, C., Rossi, F., Rustici, M., Spalla, O., Tiezzi, E.: Chemical waves and pattern formation in the 1,2-dipalmitoyl-sn-glycero-3-phosphocholine/water lamellar system. *J. Am. Chem. Soc.* **126**(37), 11406–11407 (2004)
27. Ristori, S., Rossi, F., Biosa, G., Marchettini, N., Rustici, M., Tiezzi, E.: Interplay between the Belousov-Zhabotinsky reaction-diffusion system and biomimetic matrices. *Chem. Phys. Lett.* **436**, 175–178 (2007)
28. Rossi, F., Ristori, S., Rustici, M., Marchettini, N., Tiezzi, E.: Dynamics of pattern formation in biomimetic systems. *J. Theor. Biol.* **255**(4), 404–412 (2008)
29. Torbensen, K., Rossi, F., Ristori, S., Abou-Hassan, A.: Chemical communication and dynamics of droplet emulsions in networks of Belousov-Zhabotinsky micro-oscillators produced by microfluidics. *Lab Chip* **17**(7), 1179–1189 (2017)
30. Tomasi, R., Noel, J.M., Zenati, A., Ristori, S., Rossi, F., Cabuil, V., Kanoufi, F., Abou-Hassan, A.: Chemical communication between liposomes encapsulating a chemical oscillatory reaction. *Chem. Sci.* **5**(5), 1854–1859 (2014)
31. Rossi, F., Zenati, A., Ristori, S., Noel, J.M., Cabuil, V., Kanoufi, F., Abou-Hassan, A.: Activatory coupling among oscillating droplets produced in microfluidic based devices. *Int. J. Unconventional Comput.* **11**(1), 23–36 (2015)
32. Torbensen, K., Ristori, S., Rossi, F., Abou-Hassan, A.: Tuning the chemical communication of oscillating microdroplets by means of membrane composition. *J. Phys. Chem. C* **121**(24), 13256–13264 (2017)
33. Nii, T., Ishii, F.: Properties of various phosphatidylcholines as emulsifiers or dispersing agents in microparticle preparations for drug carriers. *Colloids Surf. B: Biointerfaces* **39**(1), 57–63 (2004)
34. Di Cola, E., Torbensen, K., Clemente, I., Rossi, F., Ristori, S., Abou-Hassan, A.: Lipid stabilized water- oil interfaces studied by micro focusing small angle X-ray scattering. *Langmuir* **33**(36), 9100–9105 (2017)
35. Utada, A.S., Lorenceau, E., Link, D.R., Kaplan, P.D., Stone, H.A., Weitz, D.A.: Monodisperse double emulsions generated from a microcapillary device. *Science* **308**(5721), 537–541 (2005)
36. Stockmann, T.J., Noël, J.M., Ristori, S., Combellas, C., Abou-Hassan, A., Rossi, F., Kanoufi, F.: Scanning electrochemical microscopy of Belousov-Zhabotinsky reaction: how confined oscillations reveal short lived radicals and auto-catalytic species. *Anal. Chem.* **87**(19), 9621–9630 (2015)
37. Pikovsky, A.S., Rosenblum, M.G., Osipov, G.V., Kurths, J.: Phase synchronization of chaotic oscillators by external driving. *Phys. D: Nonlinear Phenom.* **104**(3–4), 219–238 (1997)
38. Fukuda, H., Morimura, H., Kai, S.: Global synchronization in two-dimensional lattices of discrete Belousov-Zhabotinsky oscillators. *Phys. D: Nonlinear Phenom.* **205**(1–4), 80–86 (2005)
39. Vanag, V.K., Epstein, I.R.: Excitatory and inhibitory coupling in a one-dimensional array of Belousov-Zhabotinsky micro-oscillators: theory. *Phys. Rev. E Stat. Nonlinear Soft Matter Phys.* **84**(6 Pt 2), 066209 (2011)
40. Torbensen, K., Rossi, F., Pantani, O.L., Ristori, S., Abou-Hassan, A.: Interaction of the Belousov-Zhabotinsky reaction with phospholipid engineered membranes. *J. Phys. Chem. B* **119**(32), 10224–10230 (2015)
41. Benini, O., Cervellati, R., Fetto, P.: Experimental and mechanistic study of the bromomalonic acid/bromate oscillating system catalyzed by $[\text{Fe}(\text{phen})_3]^{2+}$. *Int. J. Chem. Kinet.* **30**(4), 291–300 (1998)

42. Rossi, F., Varsalona, R., Liveri, M.L.T.: New features in the dynamics of a ferrioin-catalyzed Belousov-Zhabotinsky reaction induced by a zwitterionic surfactant. *Chem. Phys. Lett.* **463**(4–6), 378–382 (2008)
43. Rossi, F., Lombardo, R., Sciascia, L., Sbriziolo, C., Liveri, M.L.T.: Spatio-temporal perturbation of the dynamics of the ferrioin catalyzed Belousov-Zhabotinsky reaction in a batch reactor caused by sodium dodecyl sulfate micelles. *J. Phys. Chem. B* **112**, 7244–7250 (2008)
44. Noyes, R.M., Field, R., Koros, E.: Oscillations in chemical systems. I. Detailed mechanism in a system showing temporal oscillations. *J. Am. Chem. Soc.* **94**(4), 1394–1395 (1972)
45. Zhang, J., Unwin, P.R.: Kinetics of bromine transfer across Langmuir monolayers of phosphatidylethanolamines at the water/air interface. *Phys. Chem. Chem. Phys.* **5**(18), 3979–3983 (2003)
46. Xiang, T.X., Anderson, B.D.: Permeability of acetic acid across gel and liquid-crystalline lipid bilayers conforms to free-surface-area theory. *Biophys. J.* **72**(1), 223–237 (1997)
47. Kummer, U., Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P.: COPASI-a COMplex PATHway SIMulator. *Bioinformatics* **22**(24), 3067–3074 (2006)



Controlling Chemical Chaos in the Belousov-Zhabotinsky Oscillator

Marcello A. Budroni¹(✉), Mauro Rustici¹, Nadia Marchettini²,
and Federico Rossi³

¹ Dipartimento di Chimica e Farmacia, Università di Sassari,
Via Vienna 2, 07100 Sassari, Italy
mabudroni@uniss.it

² Department of Earth, Environmental and Physical Sciences - DEEP Sciences,
University of Siena, Siena, Italy

³ Department of Chemistry and Biology “A. Zambelli”,
University of Salerno, Fisciano, Italy
frossi@unisa.it

<http://physchem.uniss.it/cnl.dyn/budroni.html>

Abstract. Chaos is ubiquitous in Nature and represents one of the most fascinating expressions of real world complexity. Depending on the specific context, the onset of chaotic behaviours can be undesirable, thus, controlling the mechanisms at the basis of chaotic dynamics represents a cutting-edge challenge in many areas, including cardiology, information processing, hydrodynamics and optics, to name a few. In this work we review our recent results showing how, in chemical reactions, the active interplay between a nonlinear kinetics and hydrodynamic instabilities can be exploited as a general mechanism to induce and control chemical chaos. To this end, we consider as a model system the Belousov-Zhabotinsky (BZ) reaction. Thanks to a chemo-hydrodynamic coupling, the reaction can undergo chaotic oscillations when carried out in batch conditions. Chaos appears and disappears by following Ruelle-Takens-Newhouse scenario both in the cerium- and ferroin-catalyzed BZ systems. Here, we present experimental evidence that the transition to chemical chaos can be directly controlled by tuning either kinetic or hydrodynamic parameters of the system. Experiments were simulated by using a reaction-diffusion-convection (RDC) model where the nonlinear reaction kinetics are coupled to the Navier-Stokes equations. Numerical solutions of the RDC model clearly indicate that natural convection can feedback on the spatio-temporal evolution of the concentration fields and, in turn, changes bulk oscillation patterns. Distinct bifurcations in the oscillation patterns are found when the Grashof numbers (governing the entity of convective flows into the system) and the diffusion coefficients of the chemical species are varied. The consumption of the initial reagents is also found to be a critical phenomenon able to modulate the strength of the RDC coupling and drive order-disorder transitions.

Keywords: Chaos control · Belousov-Zhabotinsky reaction
Chemical chaos · Ruelle-Takens-Newhouse transition
Reaction-diffusion-convection model · Chemo-hydrodynamics

1 Introduction

The term “chaos” identifies deterministic aperiodic behaviours sensitive to initial conditions [1]. This means that the same chaotic system will evolve in two exponentially divergent stories when starting from two infinitesimally different initial conditions. Also popular among non-scientists as *butterfly effect*, this feature implies the long-term unpredictability of chaotic systems; in fact, though these systems are governed by deterministic rules, their macroscopic initial states cannot be known with infinite precision. In this framework, we can include the failure of the economic and weather forecasts and we can also understand why the onset of chaos is often considered undesirable, such as in the case of transitions from regular rhythm to chaotic electrical activity in the cardiac tissue, which preludes to ventricular fibrillations. In contrast, in several contexts chaos turns to be useful. An example is the realm of artificial intelligence where the complexity of chaotic sequences can be exploited as a source of information to develop fundamental logics [2,3] or to encrypt messages [4]. Independently of the context, it is always desirable to understand and control chaotic instabilities and their underlying mechanisms.

In this perspective chemical systems have traditionally played a key role. In particular, chemical oscillators, whereby the concentration of some intermediates of the reaction changes periodically in time, have been widely used as relatively simple model systems for studying chaotic dynamics [5]. The Belousov-Zhabotinsky (BZ) reaction [6,7] is the prototype of chemical oscillators. It consists of a mixture of a bromate salt, an oxidizable substrate (malonic acid in the most common recipe) and a redox catalyst (typically ferroin or cerium complexes) in a strongly acidic medium. The reaction proceeds easily at room temperature and pressure and it can stay far-from-equilibrium for a long time thanks to the slow depletion of the reactants. When stirred, the reaction shows periodic oscillations between the reduced and the oxidized state of the catalyst (and other intermediates); if the same solution is poured into a Petri dish forming a shallow layer, oxidation waves (concentric or spiral waves) periodically form and develop through the medium as a result of the spatial synchronization and spreading of the chemical oscillations driven by diffusion (see an example in Fig. 1b).

A minimal kinetic scheme that can describe the BZ oscillatory mechanism is the FKN model [5,8]. According to this scheme there are 3 fundamental processes that cyclically alternate during the reaction. The first two steps involve the depletion of bromide ions (Br^-) and the autocatalytic production HBrO_2 that, in turn, oxidises the catalyst. In the third step (the reset of the clock), the catalyst is brought back to the reduced form *via* a reaction with the organic species of the system (typically malonic acid) and, simultaneously, new Br^- ions are produced. The switching among the three steps is ruled by the concentration

of bromides, that alternatively crosses a threshold dictated by the experimental conditions (reactants concentration, temperature, etc.) and initiate either the oxidative or the reducing process. Oscillations are visible following a chromatic periodic change from red to blue when the ferroin is used as the redox catalyst. Nevertheless, in order to follow the dynamics quantitatively, spectrophotometric or potentiometric recordings are the most convenient techniques. Large-amplitude periodic oscillations in the solution absorbance typically appear in the spectrophotometric recordings when the solution is well-stirred. However, it was found that, if stirring is stopped, periodic oscillations dynamically transform into aperiodic and eventually chaotic oscillations (see Fig. 1a) [9, 10]. This phenomenon can be reproduced in a wide range of conditions and represents a sort of fingerprint of the system. Chaotic oscillations occur and vanish by following a Ruelle-Takens-Newhouse (RTN) route, that involves a sequence of Hopf bifurcations going from periodicity to quasi-periodicity and eventually to chaos [11].

The physical basis of the onset of chaos was found to be an active interplay between the nonlinear kinetics and transport phenomena (typically diffusion and convection). In fact, the nonlinear kinetics, when coupled to diffusion, induces the spontaneous formation of chemical waves that, in turn, bear concentration inhomogeneities and density gradients. In the presence of the gravitational field, unfavourable density gradients initiate buoyancy-driven convective flows that couple back with the reaction evolution and the reaction-diffusion patterns. This loop, sketched in Fig. 1c, is also called chemo-hydrodynamic coupling [12] and is known to promote complex behaviours and the formation of new stationary or dynamical patterns [13–15].

In this paper, we show how to master the system dynamics, including self-sustained chaos, by tuning the coupling between chemistry and transport phenomena. In fact, chemically-driven convection induced by an oscillatory reaction can be controlled through a simple adjustment of experimental conditions (reactants concentration, temperature, etc.) and physical properties (viscosity, reactor geometry, etc.), that act either on the kinetics or on the hydrodynamics of the system, in order to select and maintain over time a chosen dynamical reaction regime (periodic, aperiodic or chaotic). Experimental results are guided and supported by numerical simulations and interpreted in terms of a general reaction-diffusion-convection model that can be generalised and applied to similar problems.

2 Experimental Approach

2.1 Experimental

In our experiments we used both the cerium- and the ferroin-catalyzed BZ systems. Malonic acid ($\text{CH}_2(\text{COOH})_2$, MA), sodium bromate (NaBrO_3), sulfuric acid (H_2SO_4), ferroin ($\text{Fe}(\text{phen})_3^{2+}$, Fe) and cerium sulfate ($\text{Ce}(\text{SO}_4)_2$, Ce(IV)) were purchased from Sigma Aldrich. All reagents were of analytical quality and were used without further purification. Deionised water from reverse osmosis was used to prepare all the solutions.

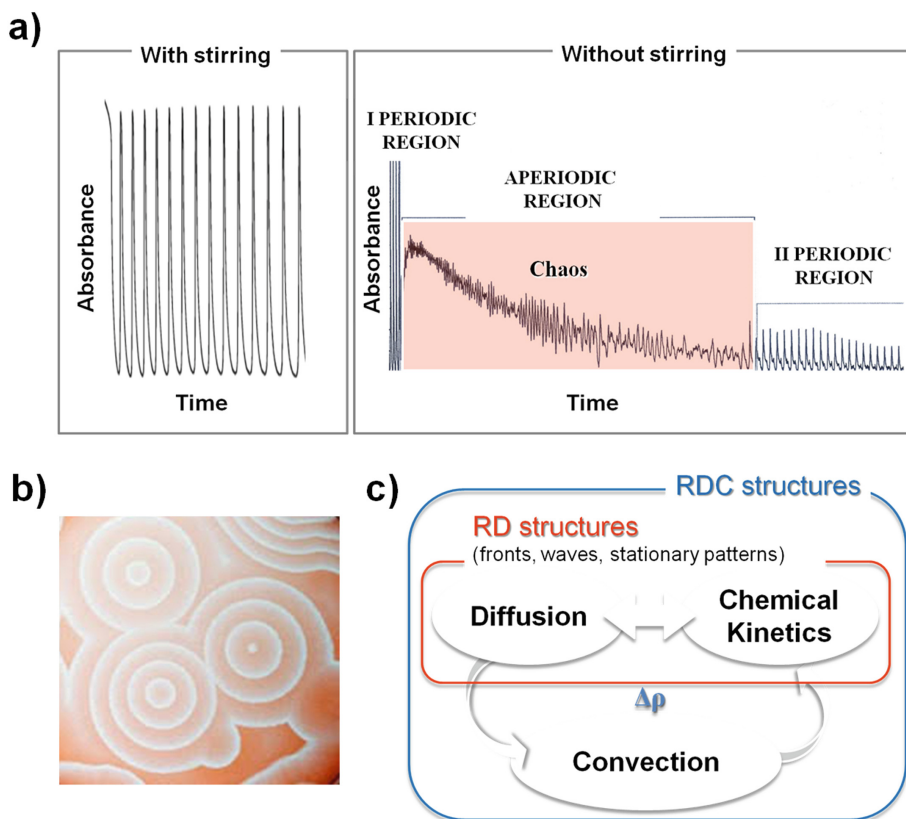


Fig. 1. (a) Examples of spectrophotometric recordings of the Ce(IV)-catalyzed BZ reaction in a batch unstirred reactor. On the left the typical oscillations characterizing a well-stirred solution while on the right the evolution of the unstirred reaction. The red box identifies the characteristic aperiodic transient between the two periodic regions. $[MA] = 0.3 \text{ M}$, $[NaBrO_3] = 0.09 \text{ M}$, $[H_2SO_4] = 1 \text{ M}$, $[Ce(IV)] = 4 \text{ mM}$. (b) Example of concentric chemical waves developing in the ferriin-catalyzed BZ medium. (c) Scheme of the complex interplay between nonlinear kinetics and transport phenomena, sustaining density-driven chemo-hydrodynamic patterns and the transition to chemical chaos. (Color figure online)

The kinetics of the BZ reaction has been studied at $25.0 \text{ }^\circ\text{C}$. The typical recipe for the cerium-catalyzed system was $[Ce(IV)] = 0.004 \text{ M}$, $[MA] = 0.30 \text{ M}$, $[NaBrO_3] = 0.09 \text{ M}$, $[H_2SO_4] = 1 \text{ M}$ while the following concentrations $[MA] = 0.74 \text{ M}$, $[NaBrO_3] = 0.28 \text{ M}$, $[H_2SO_4] = 0.35 \text{ M}$, $[Fe] = 0.93 \text{ mM}$ were used for the ferriin-catalyzed system.

The reaction dynamics was monitored by recording *via* a UV-vis spectrophotometer the absorption of (i) Ce(IV) for the cerium-catalyzed system at $\lambda_{max} = 320 \text{ nm}$ ($\epsilon \sim 5600 \text{ M}^{-1} \text{ cm}^{-1}$) and (ii) the ferriin, the oxidized form of ferriin, at $\lambda_{max} = 630 \text{ nm}$ ($\epsilon \sim 620 \text{ M}^{-1} \text{ cm}^{-1}$) for the ferriin-catalyzed BZ

system. 3.0 mL of the reactive solution were prepared in a beaker, stirred for twenty minutes and finally transferred into a quartz cuvette for spectrophotometric data acquisition. Each kinetic measurement has been repeated at least three times in order to check the reproducibility of the experimental results. The time series obtained in this way were analyzed by means of the Fast Fourier Transform (FFT).

2.2 Hydrodynamic Control

Influence of Stirring. An immediate and straightforward control over the chemo-hydrodynamic coupling responsible for chaos, is to restart stirring during the development of the aperiodic transient [10,16]. In this way we can suppress the onset of natural convection as we eliminate the concentration gradients at the origin of the buoyancy-driven hydrodynamic instabilities. When stirred, the system behaves as a unique oscillator with regular high-amplitude oscillations. If stirring is stopped again, the system dynamics undergoes a new transition to the aperiodic regimes. This is illustrated in Fig. 2. We expect that a similar behaviour can also be obtained if the reaction is carried-out in parabolic flights (see as an example the experiments run in microgravity with the Iodide-Arseneous-Acid (IAA) reaction [17]), where periodic conditions of microgravity eliminate and decouple intermittently the contribution of buoyancy-driven convection to the system dynamics.

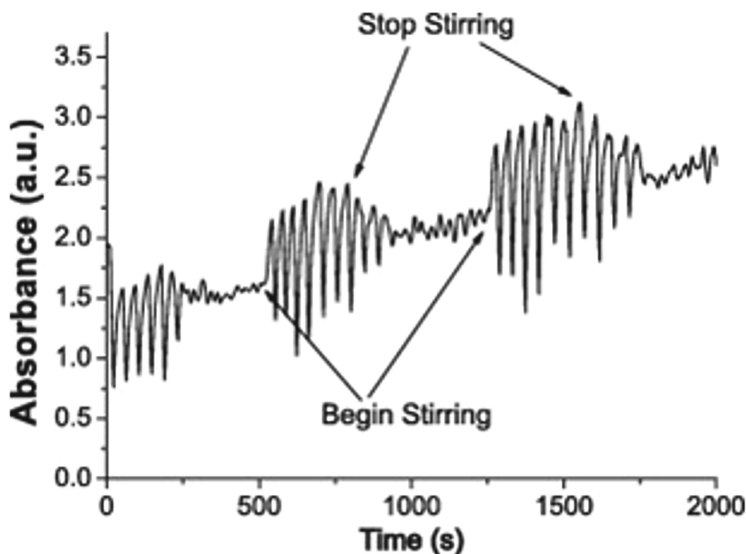


Fig. 2. Effect of stirring when ferroin catalysed BZ reaction undergoes a transition to chemical chaos. Reproduced from [10] with permission of the copyright owner.

Influence of the Reactor Size. Hydrodynamic instabilities are also known to be sensitive to the size of the spatial domain where they occur and can be avoided working with reactors below a critical size. A spectrophotometric study on the dynamic behaviour of the BZ system in unstirred batch conditions was then carried out by using cuvettes with different path length, specifically in the range 1 to 0.02 cm [18]. It was shown that there is a critical threshold, namely 0.05 cm, below which the transition to aperiodic oscillations cannot be observed any more and just periodic oscillations develop as the possibility for the onset of convection is also hindered by narrowing the reactor (see Fig. 3).

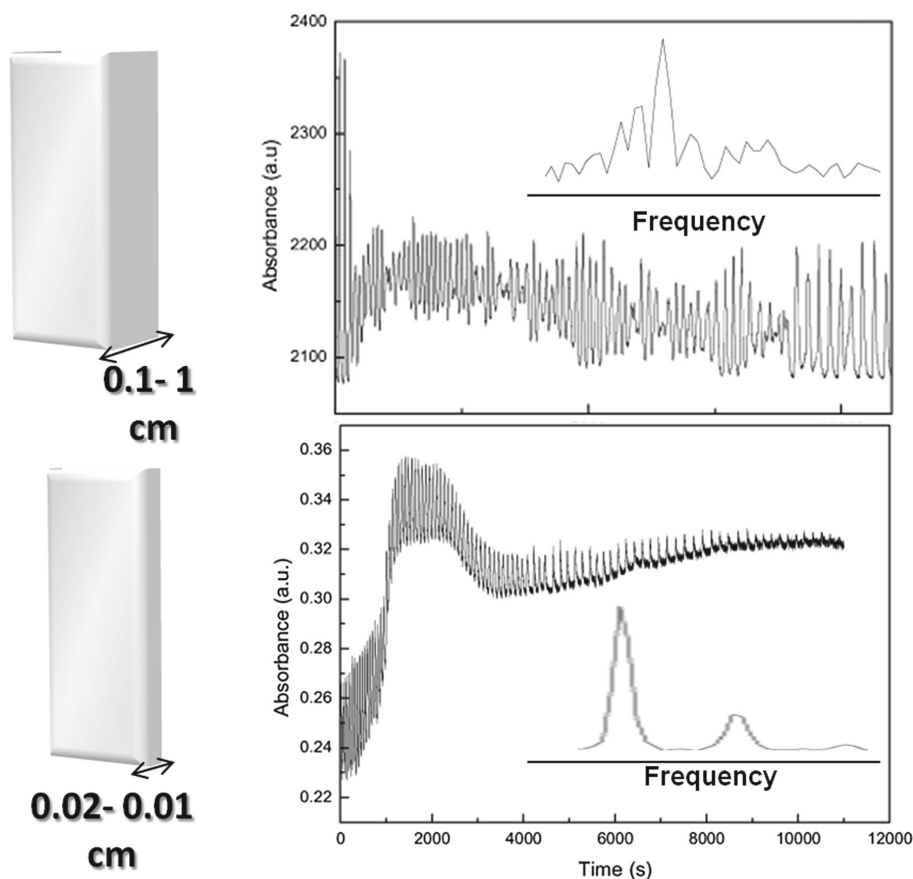


Fig. 3. Effect of the reactor size in the dynamics of the BZ reaction in batch and unstirred reactors.

Influence of the Medium Viscosity. A further control of the system dynamics can be obtained by changing the medium viscosity. In our check experiments this was obtained for example by adding different amounts of an organic polymer, namely poly-ethylene-glycol (PG), to the reactive solution [16] or by using

a surfactant, sodium dodecyl sulphate (SDS) [19,20], both able to increase the hydrodynamic inertia of the medium to contrast convective motions without affecting the chemical kinetics. In particular, differently to the case of the zwitterionic surfactant N-tetradecyl-N,N-dimethylamine oxide that causes an induction period prior to the onset of regular oscillations [19,21], SDS only slightly alter the kinetics of the BZ reaction, without changing the oscillation mechanism and without introducing new dynamical features [22], even above the critical micelle concentration (CMC). To maximise the effect of the surfactant on the viscosity of the solutions, we thus varied SDS concentration above CMC.

Both PG and SDS, by suppressing the possibility for the onset of hydrodynamic motions, can in parallel prevent the system from a transition to chemical chaos. The percentage of PG and the concentration of SDS are related to the kinematic viscosity and this can effectively act as a direct control parameter in the bifurcation sequence from chaotic to periodic regimes. Once more, it was found that this route to periodicity obeys a RTN scenario. To give an example of this result, we report in Fig. 4 the transition scenario from chaos to periodicity obtained by increasing the concentration of SDS in the range [1, 250] mM, which causes an increase of viscosity up 10% as compared to the surfactant-free BZ system.

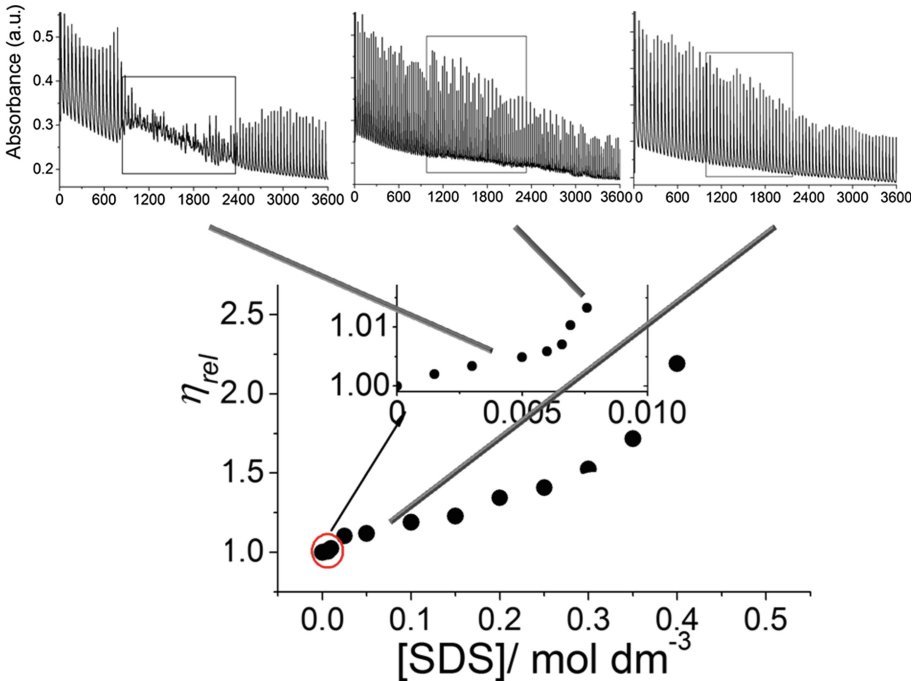


Fig. 4. Relative viscosity, η_{rel} , of the ferroin-catalyzed BZ solution and related effect in the chemical oscillator dynamics when the reaction is carried out in batch conditions and without stirring. $[MA] = 0.74$ M, $[NaBrO_3] = 0.28$ M, $[H_2SO_4] = 0.35$ M; the inset shows the zoom of the region $0 < [SDS] < 1 \times 10^{-2}$ M.

2.3 Chemical Control

In order to show that not only hydrodynamics but also chemistry plays a crucial role in the appearance of chemical chaos, a large number of experiments were carried out by varying different relative initial concentration of the reactants of the BZ oscillator. As shown by Pojman et al. [23], the concentration of the main reactants can be treated as a pseudo-bifurcation parameter in batch conditions. A systematic screening of the ternary parameter-space was performed [24] by varying the relative concentration of the redox catalyst (here CeSO_4), NaBrO_3 and malonic acid. It was found that by changing the initial composition of the system the oscillatory dynamics and, hence, the transition to chaos, could be controlled to a large extent. This is illustrated in the ternary diagram shown in Fig. 5. When the relative concentration of the reactants is that circumscribed by the red region, chaos can appear after the periodic behaviour following a RTN route. Yellow areas indicate conditions where only quasi-periodicity can develop and, finally, green region are related to situations in which only periodic behaviours were observed.

The ionic strength of the solution is a further tool to control the dynamics of the system through chemistry. In fact, it was demonstrated that by adding an inert electrolyte to the solution (Na_2SO_4 , $\text{Al}_2(\text{SO}_4)_3$, etc.) the chemical potential of the reactants could be tuned to prevent or induce the chaotic regime [25].

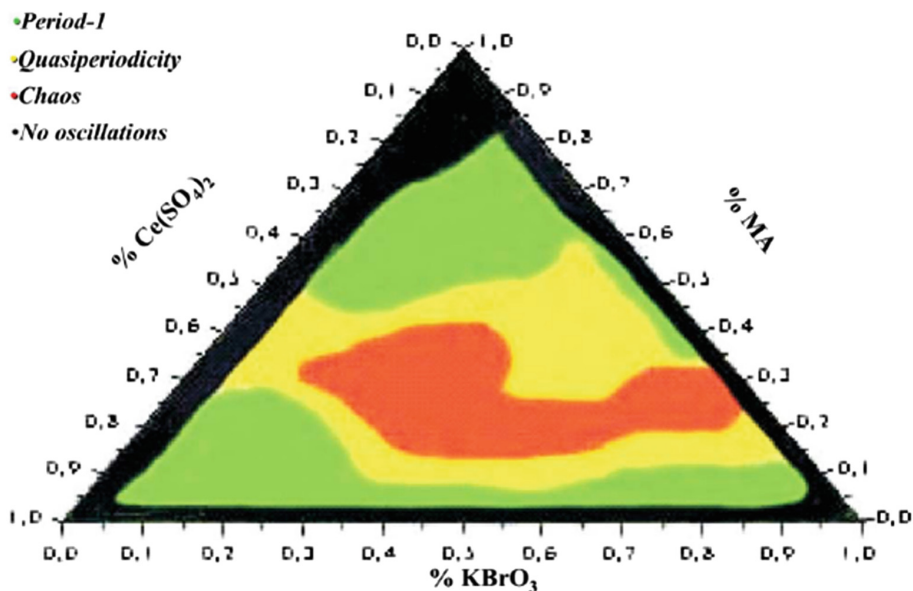


Fig. 5. A ternary bifurcation diagram describing possible dynamical regimes in a $\text{Ce}(\text{IV})$ -catalyzed closed unstirred BZ system as a function of the volume fraction of three initial reactants: malonic acid, potassium bromate and $\text{Ce}(\text{IV})$. The green, yellow and red zones identify periodic, quasi-periodic and chaotic domains, respectively. The black boundary zone corresponds to initial compositions where no oscillations occur. Reproduced from [24] with permission of the copyright owner. (Color figure online)

3 Numerical Approach

The experimental approach discussed so far could be strengthened by means of a theoretical/numerical implementation. In fact, the modelling strategy helped us in the interpretation of the experimental results and now serves as powerful planning instrument to predict new routes for chaos control.

3.1 Model

We modeled the system as a two-dimensional vertical slab (i.e. a vertical cut of the real three-dimensional spectrophotometric cuvette, perpendicular to a virtual spectrophotometric beam) in the coordinate system (x, y) , with the gravitational field $\mathbf{g} = (0, -g)$ oriented against the vertical axis y . As shown in previous work, this two-dimensional description is a reliable approximation to the three-dimensional problem [26–29]. A set of reaction-diffusion-convection (RDC) equations is derived by coupling the chemical kinetics to diffusion through Fick's terms and to natural convection by means of the Navier-Stokes equations.

The reaction-diffusion-convection (RDC) system is (i) formulated in the Boussinesq approximation, (ii) written in the vorticity-stream function $(\omega - \psi)$ form, (iii) conveniently scaled on the chemical time scale t_0 (see [5]) and on the characteristic space scale of the problem, x_0 . Finally, since the BZ reaction is not highly exothermic and thermal gradients are rapidly smoothed, we formulated the problem under the isothermal approximation.

The resulting model is

$$\frac{\partial c_i}{\partial t} + D_\nu \left(u \frac{\partial c_i}{\partial x} + v \frac{\partial c_i}{\partial y} \right) - D_i \nabla^2 c_i = k_i(c_i, \bar{\lambda}) \quad i = 1, 2 \quad (1)$$

$$\frac{\partial \omega}{\partial t} + D_\nu \left(u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} \right) - D_\nu \nabla^2 \omega = -D_\nu \sum_i Gr_i \frac{\partial c_i}{\partial x} \quad (2)$$

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = -\omega \quad (3)$$

$$u = \frac{\partial \psi}{\partial y} \quad (4)$$

$$v = -\frac{\partial \psi}{\partial x} \quad (5)$$

$D_\nu = \nu t_0 / x_0^2 = 58.5$ is the dimensionless viscosity (ν being the medium kinematic viscosity); $D_i = D t_0 / x_0^2 = 0.00350$ is the dimensionless diffusivity (D being the dimensional diffusivity of the two oscillating species). $u = U/v_0$ and $v = V/v_0$ are dimensionless horizontal and vertical components of the velocity field scaled over the velocity scale $v_0 = x_0/t_0$.

$Gr_i = g x_0^3 \delta \rho_i / \rho_0 \nu^2$ is the Grashof number for the i -th species, g is the gravitational acceleration, ρ_0 is the reference density of the medium and $\delta \rho_i = \frac{\partial \rho}{\partial c_i}$ is

the density variation due to the change of the concentration of the i -th species with respect to the reference conditions (reduced state) of the reactive mixture. The Grashof number is a measure of the sensitivity of a species to produce convective motions in virtue of isothermal density changes and, in a sense, controls the strength of the RDC coupling within the system. ω is the vorticity, defined as the rotor of the velocity vector (u, v) , while the stream function ψ is defined by Eqs. (4) and (5).

The kinetic functions $k_i(c_i, \bar{\lambda})$ are derived from the *Oregonator* model [5] and have the form

$$k_1(c_i, \bar{\lambda}) = \frac{dc_1}{d\tau} = \frac{1}{\epsilon_1} \left(\frac{(qa - c_1)}{(qa + c_1)} fbc_2 + c_1(a - c_1) \right) \quad (6)$$

$$k_2(c_i, \bar{\lambda}) = \frac{dc_2}{d\tau} = ac_1 - bc_2 \quad (7)$$

where $i = 1, 2$, c_1 is the concentration of bromous acid, c_2 is the concentration of the oxidized form of the catalyst and $\bar{\lambda} = \epsilon_1, q, f, a, b$ the set of kinetic parameters. The initial distributions of the chemical species are set as

$$c_1(0) = 0.8 \text{ if } 0 < \theta < 0.5 \quad (8)$$

$$= c_{1(ss)} \text{ elsewhere} \quad (9)$$

$$c_2(0) = c_{2(ss)} + \frac{\theta}{8\pi f} \quad (10)$$

(where $c_{2(ss)} = c_{1(ss)} = q(f + 1)/(f - 1)$ and θ is the polar coordinate angle) to mimic inhomogeneous concentration profiles that typically occur in unstirred systems. These specific functions were used by Jahnke et al. [30] to initiate spiral waves in an analogous reaction-diffusion system. q and ϵ are kinetic parameters accounting for the excitability of the system and f is a stoichiometric factor included in the resetting step of the oscillatory scheme. This parameter allows one to set the system in an oscillatory regime when it ranges $[0.5, 1 + \sqrt{2}]$ and we use $f = 1.6$; q is fixed to 0.01; $\epsilon = 0.01$; a and b are the concentration of the bromate salt and the malonic acid, respectively. In our study we set $b = 1$ while a was used a chemical control parameter.

The PDE system (1–5) was numerically solved over a 100×100 points grid (mesh-point separation $h_x = 0.50$), using the alternating direction finite difference method [31]. We imposed no-slip boundary conditions for the fluid velocity and no-flux boundary conditions for chemical concentrations at the walls of the slab. A small time step h_t has to be used due to the stiff nature of the kinetic equations. $h_t = 1 \times 10^{-6}$ was tested to be a good value.

In the experiments the output of the spectrophotometric recordings is the average of the absorbance of the reactive solution over the spatial domain scanned by the spectrophotometric beam as a function of the time. In order to have an observable comparable to the experimental data, we build up time series by reporting at each integration time step the mean concentration of the oscillatory intermediates averaged over the solving grid ($\langle c_1 \rangle$ and $\langle c_2 \rangle$). The resulting signals are then analyzed by means of the FFT and attractor's reconstruction.

3.2 Hydrodynamic Control

We focus now on the direct transition from periodic to chaotic regimes under hydrodynamic control, namely by changing the Grashof numbers of the chemical species [26]. This gives a picture of the direct transition to chaos in the unstirred BZ reaction (shown in Fig. 1) if one assumes that, after stirring is stopped, residual advective motions relax, concentration patterns (typically waves) with related density gradients can build-up and initiate convective motions with progressively growing intensity. In this regime the chemical oscillator is in the far-from-equilibrium branch, where the depletion of the initial reactants is negligibly slow and the system can maintain quasi-stationary conditions like if it was open.

Periodic regime. In Fig. 6a we show a limit cycle, with a fundamental frequency $\omega_1 = 0.747$ Oregonator frequency units, obtained in the absence of convection (i.e. with $Gr_i = 0.00$). Periodicity is still found when the chemo-hydrodynamic coupling is strengthened, by increasing both Grashof numbers to 9.40. The related attractor projection ($\langle c_1 \rangle$, $\langle c_2 \rangle$) and the time series are shown in Fig. 6(b), left and centre panels, respectively. To show the attractor change, we keep fixed the region framing the phase portrait. The oscillatory dynamics of $\langle c_1 \rangle$ and $\langle c_2 \rangle$ present one fundamental frequency $\omega_1 = 0.396$, different from that observed in panel (a). This confirms that convection is actively coupled with the reaction-diffusion system. Note that, due hydrodynamic inertia, the new solution presents a longer period with respect to the case where convection is at rest.

Quasi-periodic regime. As the Grashof numbers are increased to 9.80 a quasiperiodic behavior is found (see Fig. 7). This can be inferred by the attractor reconstruction (left), the time series (centre) and quantitatively revealed by the related Fourier amplitude spectrum (right). In the latter, two characteristic fundamental frequencies (ω_1 , ω_2) and their linear combinations are shown. These frequencies, which ratio ω_1/ω_2 is an irrational number, characterize the toroidal flow of the system represented in the phase-space projection ($\langle c_1 \rangle$, $\langle c_2 \rangle$).

Chaotic regime. If the Grashof numbers are further increased, namely to 12.10, an aperiodic behavior (see Fig. 8a) associated with the strange attractor in Fig. 8c, is observed. As shown in Fig. 8a, the time series manifest sensitivity to initial conditions consistent with one of the signatures for chaotic dynamics. To test for chaos, we have also calculated the largest Lyapunov exponents, λ , using the Rosenstein algorithm from TISEAN package [32]. The value $\lambda = 0.018$ was extracted from linear regression of the curves $S(\epsilon, m, t)$ for $m = 5-9$ shown Fig. 8d.

3.3 Chemical Control

As mentioned in Sect. 2.3, we can use the initial concentration of reactants as a bifurcation *pseudo-parameter* [23, 24] to modify and control chemically the system dynamics. An inverse transition chaos-periodicity consistent with a RTN scenario was indeed induced keeping constant the Grashof numbers and decreasing the sodium bromate concentration, i.e. parameter a [33]. The chaotic regime, occurring for $a = 1$, has been extensively characterized in Sect. 3.2 and [26].

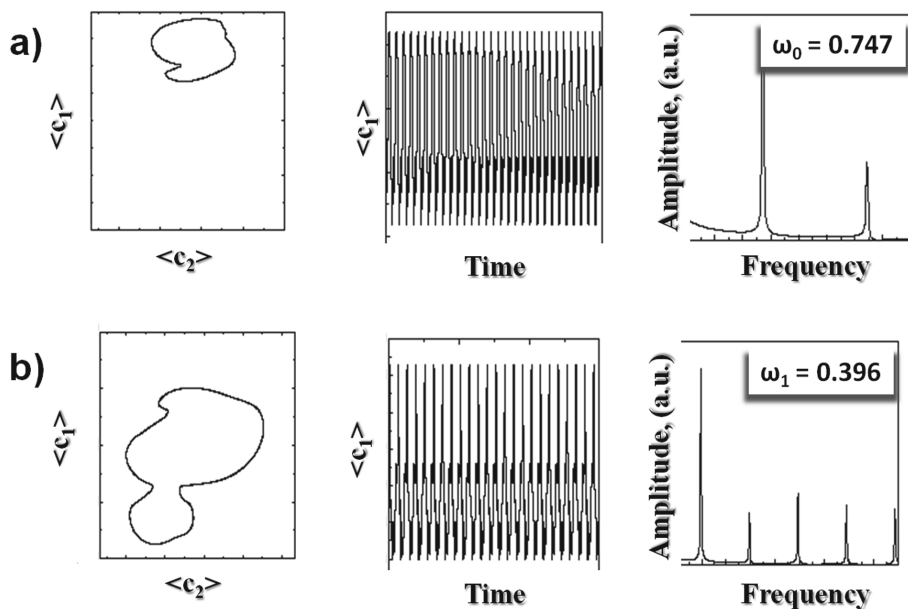


Fig. 6. (a) Characterization of the periodic dynamics of the RDC system for $Gr_1 = Gr_2 = 0.00$: (left) trajectories described by the system in the phase space projection ($\langle c_1 \rangle, \langle c_2 \rangle$) ($\langle c_1 \rangle \in [0.02, 0.20]$ and $\langle c_2 \rangle \in [0.06, 0.12]$); (centre) time series describing $\langle c_2 \rangle$ dynamics in the time-frame [100, 200] Oregonator time units; (right) FFT amplitude spectrum. (b) The same analysis is performed for $Gr_1 = Gr_2 = 9.40$.

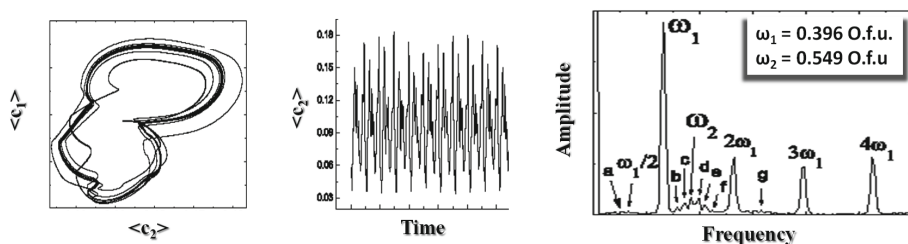


Fig. 7. Quasi-periodic regime for $Gr_1 = Gr_2 = 9.80$: phase space trajectories in the phase space projection ($\langle c_1 \rangle, \langle c_2 \rangle$) with $\langle c_1 \rangle \in [0.02, 0.20]$ and $\langle c_2 \rangle \in [0.06, 0.12]$ (left), time series describing $\langle c_2 \rangle$ dynamics in the time-frame [100:200] Oregonator time units (centre) and related Fourier amplitude spectrum (right) (a = $\omega_2 - \omega_1$, b = $3\omega_2 - 3\omega_1$, c = $6\omega_2 - 7\omega_1$, d = $\omega_1 + 1/2\omega_1$, e = $4\omega_2 - 4\omega_1$, f = $7\omega_2 - 8\omega_1$, g = $\omega_1 + \omega_2$).

A bifurcation to quasi-periodicity takes place for $a = 0.97$ and it is characterized in Fig. 9(a, b). Quasi-periodicity is confirmed by the Fourier amplitude spectrum, showing two incommensurable fundamental frequencies ($\omega_1 = 0.39$, $\omega_2 = 0.54$) and their harmonic combinations. Note that these frequencies match

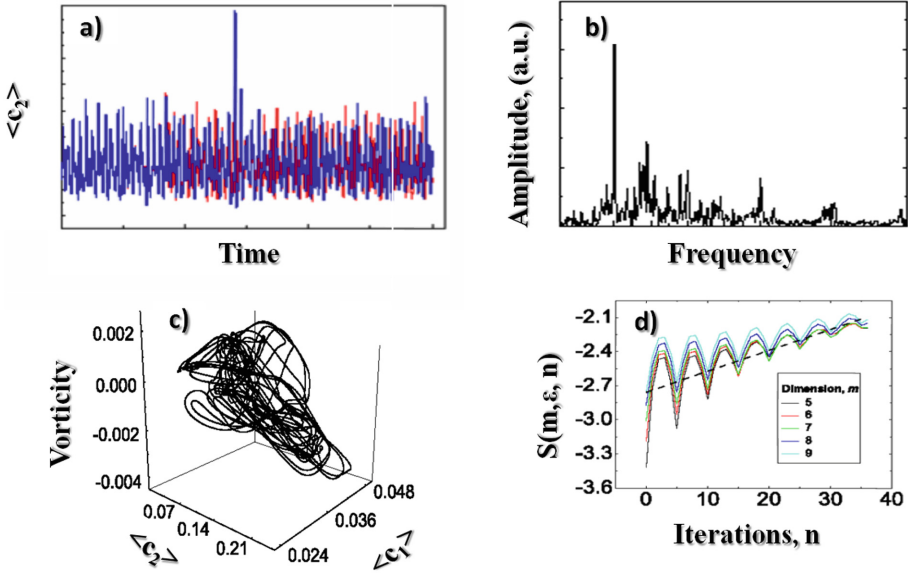


Fig. 8. Chaotic dynamics for $Gr_1 = Gr_2 = 12.10$: (a) time series describing $\langle c_2 \rangle$ evolutions for two different initial conditions in the time-frame [100:200] Oregonator time units; (b) FFT amplitude spectrum of the signal in panel (a); (c) strange attractor of this chaotic regime obtained in the phase-space ($\langle c_1 \rangle$, $\langle c_2 \rangle$, vorticity); (d) Computation of the maximum Lyapunov exponent by means of the Rosestein algorithm. The value of $\lambda = 0.018$ is obtained by the linear regression of the curves $S(\epsilon, m, t)$ for $m = 5-9$, in the zone between 0–40 iterations.

the values $\omega_1 = 0.39$, $\omega_2 = 0.54$ of the quasi-periodic regime obtained in the direct transition under hydrodynamic control.

When a is decreased to 0.95 a supercritical Hopf bifurcation leading to a bi-periodic solution can be detected. In the corresponding FFT's amplitude spectrum (Fig. 9c), the main frequency ω_1 can be still identified and subharmonic frequencies of the type $n \times \frac{\omega_1}{2}$ (where n is an integer) clearly emerge. According to the FFT's spectrum, the corresponding attractor exhibits a double-period, visible in the inset of Fig. 9d.

Figure 9f shows a limit cycle characterized by the main frequency ω_1 (see the related FFT spectrum in Fig. 9e) obtained when a reaches the value 0.93. The FFT's analysis reveals a supercritical Pichfork bifurcation, leading to a unique oscillation period.

As a whole, this transition scenario under chemical control can describe the inverse route from chaos to periodicity observed in Fig. 1a.

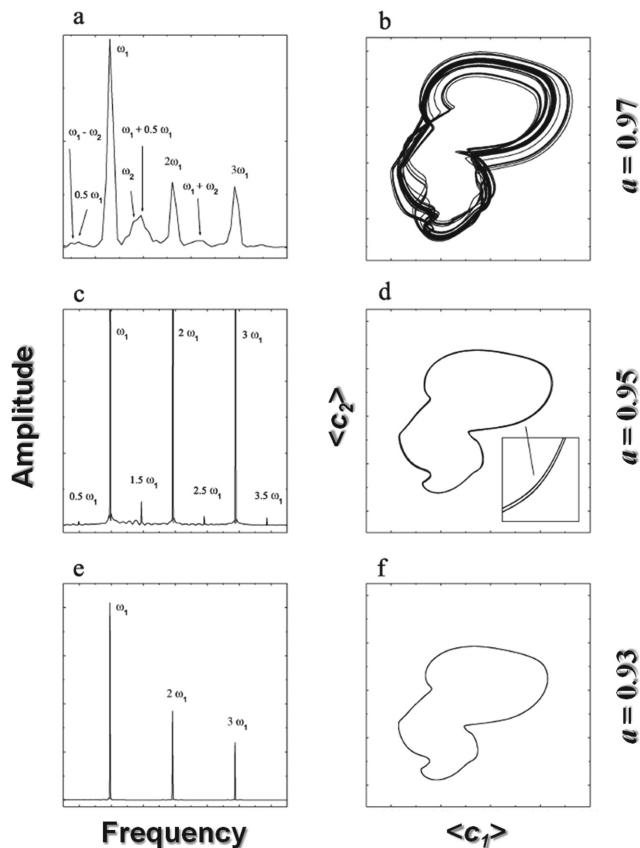


Fig. 9. Attractor reconstruction in the phase-space section $(\langle c_1 \rangle, \langle c_2 \rangle)$, with $\langle c_1 \rangle \in [0.02, 0.20]$, $\langle c_2 \rangle \in [0.06, 0.12]$ and FFT amplitude spectra of the simulated dynamical regimes in the transition from chaotic to periodic oscillations controlled by the concentration of the initial reactant a . (a–b), $a = 0.97$ a quasi-periodic regime; (c–d), $a = 0.95$ a bi-periodic regime; (e–f), $a = 0.93$ a periodic regime.

4 Concluding Discussion

To summarize, we discussed the active interplay of nonlinear kinetics with related chemically-driven transport phenomena as a general mechanism for devising a self-sustained chaotic generator. The route to chaos in this context can be controlled by tuning the strength of the chemo-hydrodynamic coupling either *via* chemical or hydrodynamic parameters. We have supported this idea by means of experimental examples and also formalized it with a reaction-diffusion-convection model which allows the numerical description and prediction of the chaotic dynamics.

This theoretical framework guided us in the interpretation of the transition from periodicity to chaos and *viceversa* observed in experiments. In particular,

it was found that the reaction evolves through two main phases, characterized by a longer time scale with respect to chemical oscillations. In a first phase, the concentration of the reactants is in large excess with respect to the intermediates and the reactant depletion can be neglected. In this phase, the system is mainly under hydrodynamic control and convection drives the system to chaos. In a second phase, the system evolves to the ultimate thermodynamic equilibrium. The main reactants consumption cannot be neglected any more and the reactants concentration acts as a bifurcation parameter towards regular periodic oscillations.

Conceptually, the results obtained with this experimental system and the related theoretical model have a general value and can also be extended to spatiotemporal phenomena [34]. The modularity of the RDC model permits to fit our findings to isomorphic problems; by changing the kinetics terms, for example, we can study other nonlinear chemical systems or face more complex mechanisms such those that lead and control low-dimensional spatio-temporal turbulence in cardiac arrhythmias.

Also, chaotic dynamics are themselves rich sources of information [4]. In the realm of artificial intelligence chemo-hydrodynamic systems could be thus exploited as generators of chaotic signals for implementing fundamental logics and as a controllable contaminator in protocols for encrypting messages. Similar studies have already been initiated by using externally-forced chemo-hydrodynamic oscillations, which feature suitable output to develop fundamental operations based on fuzzy logic [2,3].

Acknowledgments. FR gratefully acknowledges the University of Salerno for the grants ORSA158121 and ORSA167988. MAB and MR acknowledge financial support from Fondazione Banco di Sardegna.

References

1. Abarbanel, H.D.I.: Analysis of Observed Chaotic Data. Springer, New York (1996). <https://doi.org/10.1007/978-1-4612-0763-4>
2. Hayashi, K., Gotoda, H., Gentili, P.L.: Probing and exploiting the chaotic dynamics of a hydrodynamic photochemical oscillator to implement all the basic binary logic functions. *Chaos Interdisc. J. Nonlinear Sci.* **26**(5), 053102 (2016)
3. Gentili, P.L., Giubila, M.S., Heron, B.M.: Processing binary and fuzzy logic by chaotic time series generated by a hydrodynamic photochemical oscillator. *ChemPhysChem* **18**(13), 1831–1841 (2017)
4. Boccaletti, S., Grebogi, C., Lai, Y.C., Mancini, H., Maza, D.: The control of chaos: theory and applications. *Phys. Rep.* **329**(3), 103–197 (2000)
5. Scott, S.K.: *Chemical Chaos*. Oxford University Press, Oxford (1993)
6. Belousov, B.P.: A periodic reaction and its mechanism. In: *Sbornik Referatov po Radiatsionno Meditsine*, Moscow, Medgiz, pp. 145–147 (1958)
7. Zhabotinsky, A.M.: Periodic liquid phase reactions. *Proc. Acad. Sci. USSR* **157**, 392–395 (1964)
8. Noyes, R.M., Field, R., Koros, E.: Oscillations in chemical systems. I. Detailed mechanism in a system showing temporal oscillations. *J. Am. Chem. Soc.* **94**(4), 1394–1395 (1972)

9. Rustici, M., Branca, M., Caravati, C., Marchettini, N.: Evidence of a chaotic transient in a closed unstirred cerium catalyzed Belousov-Zhabotinsky system. *Chem. Phys. Lett.* **263**(3), 429–434 (1996)
10. Rossi, F., Budroni, M.A., Marchettini, N., Cutietta, L., Rustici, M., Turco Liveri, M.L.: Chaotic dynamics in an unstirred ferroin catalyzed Belousov-Zhabotinsky reaction. *Chem. Phys. Lett.* **480**(4–6), 322–326 (2009)
11. Newhouse, S., Ruelle, D., Takens, F.: Occurrence of strange Axiom A attractors near quasi periodic flows on T^m , $m \geq 3$. *Commun. Math. Phys.* **64**(1), 35–40 (1978)
12. De Wit, A., Eckert, K., Kalliadasis, S.: Introduction to the focus issue: chemo-hydrodynamic patterns and instabilities. *Chaos Interdisc. J. Nonlinear Sci.* **22**(3), 037101 (2012)
13. Rossi, F., Turco Liveri, M.L.: Chemical self-organization in self-assembling biomimetic systems. *Ecol. Model.* **220**(16), 1857–1864 (2009)
14. Rossi, F., Budroni, M.A., Marchettini, N., Carballido-Landeira, J.: Segmented waves in a reaction-diffusion-convection system. *Chaos Interdisc. J. Nonlinear Sci.* **22**(3), 037109–037109-11 (2012)
15. Budroni, M.A., De Wit, A.: Dissipative structures: From reaction-diffusion to chemo-hydrodynamic patterns. *Chaos Interdisc. J. Nonlinear Sci.* **27**(10), 104617 (2017)
16. Marchettini, N., Rustici, M.: Effect of medium viscosity in a closed unstirred Belousov-Zhabotinsky system. *Chem. Phys. Lett.* **317**(6), 647–651 (2000)
17. Horvath, D., Budroni, M.A., Baba, P., Rongy, L., De Wit, A., Eckert, K., Hauser, M.J.B., Toth, A.: Convective dynamics of traveling autocatalytic fronts in a modulated gravity field. *Phys. Chem. Chem. Phys.* **16**, 26279–26287 (2014)
18. Turco Liveri, M.L., Lombardo, R., Masia, M., Calvaruso, G., Rustici, M.: Role of the reactor geometry in the onset of transient chaos in an unstirred Belousov-Zhabotinsky system. *J. Phys. Chem. A* **107**(24), 4834–4837 (2003)
19. Budroni, M.A., Rossi, F.: A novel mechanism for in situ nucleation of spirals controlled by the interplay between phase fronts and reaction-diffusion waves in an oscillatory medium. *J. Phys. Chem. C* **119**(17), 9411–9417 (2015)
20. Budroni, M.A., Calabrese, I., Miele, Y., Rustici, M., Marchettini, N., Rossi, F.: Control of chemical chaos through medium viscosity in a batch ferroin-catalysed Belousov-Zhabotinsky reaction. *Phys. Chem. Chem. Phys.* **19**, 32235–32241 (2017)
21. Rossi, F., Varsalona, R., Turco Liveri, M.L.: New features in the dynamics of a ferroin-catalyzed Belousov-Zhabotinsky reaction induced by a zwitterionic surfactant. *Chem. Phys. Lett.* **463**(4–6), 378–382 (2008)
22. Rossi, F., Lombardo, R., Sciascia, L., Sbriziolo, C., Turco Liveri, M.L.: Spatio-temporal perturbation of the dynamics of the ferroin catalyzed Belousov-Zhabotinsky reaction in a batch reactor caused by sodium dodecyl sulfate micelles. *J. Phys. Chem. B* **112**, 7244–7250 (2008)
23. Strizhak, P.E., Kawczynski, A.L.: Complex transient oscillations in the Belousov-Zhabotinskii reaction in a batch reactor. *J. Phys. Chem.* **99**(27), 10830–10833 (1995)
24. Biosa, G., Masia, M., Marchettini, N., Rustici, M.: A ternary nonequilibrium phase diagram for a closed unstirred Belousov-Zhabotinsky system. *Chem. Phys.* **308**(1), 7–12 (2005)
25. Rossi, F., Pulselli, F., Tiezzi, E., Bastianoni, S., Rustici, M.: Effects of the electrolytes in a closed unstirred Belousov-Zhabotinsky medium. *Chem. Phys.* **313**, 101–106 (2005)

26. Budroni, M.A., Masia, M., Rustici, M., Marchettini, N., Volpert, V., Cresto, P.C.: Ruelle-Takens-Newhouse scenario in reaction-diffusion-convection system. *J. Chem. Phys.* **128**(11), 111102–111104 (2008)
27. Budroni, M.A., Masia, M., Rustici, M., Marchettini, N., Volpert, V.: Bifurcations in spiral tip dynamics induced by natural convection in the Belousov-Zhabotinsky reaction. *J. Chem. Phys.* **130**(2), 024902–8 (2009)
28. Rongy, L., Schusztter, G., Sinkó, Z., Tóth, T., Horváth, D., Tóth, A., De Wit, A.: Influence of thermal effects on buoyancy-driven convection around autocatalytic chemical fronts propagating horizontally. *Chaos Interdisc. J. Nonlinear Sci.* **19**(2), 023110 (2009)
29. Budroni, M.A., Rongy, L., De Wit, A.: Dynamics due to combined buoyancy- and Marangoni-driven convective flows around autocatalytic fronts. *Phys. Chem. Chem. Phys.* **14**, 14619–14629 (2012)
30. Jahnke, W., Skaggs, W.E., Winfree, A.T.: Chemical vortex dynamics in the Belousov-Zhabotinskii reaction and in the two-variable Oregonator model. *J. Phys. Chem.* **93**(2), 740–749 (1989)
31. Peaceman, D.W., Rachford, H.H.: The numerical solution of parabolic and elliptic differential equations. *J. Soc. Ind. Appl. Math.* **3**, 28 (1955)
32. Hegger, R., Kantz, H., Schreiber, T.: Practical implementation of nonlinear time series methods: the TISEAN package. *Chaos: Interdisc. J. Nonlinear Sci.* **9**, 413 (1999)
33. Marchettini, N., Budroni, M.A., Rossi, F., Masia, M., Turco Liveri, M.L., Rustici, M.: Role of the reagents consumption in the chaotic dynamics of the Belousov-Zhabotinsky oscillator in closed unstirred reactors. *Phys. Chem. Chem. Phys.* **12**(36), 11062–11069 (2010)
34. Agladze, K.I., Krinsky, V.I., Pertsov, A.M.: Chaos in the non-stirred Belousov-Zhabotinsky reaction is induced by interaction of waves and stationary dissipative structures. *Nature* **308**(5962), 834–835 (1984)



Fragment Based Molecular Dynamics for Drug Design

Lucia Sessa¹ , Luigi Di Biasi¹, Simona Concilio² ,
and Stefano Piotto¹ 

¹ Department of Pharmacy, University of Salerno,
Via Giovanni Paolo II, 132, 84084 Fisciano, SA, Italy
lucessa@unisa.it

² Department of Industrial Engineering, University of Salerno,
Via Giovanni Paolo II, 132, 84084 Fisciano, SA, Italy

Abstract. Molecular docking is a computationally efficient method used to predict the conformations adopted by the ligand within a target-binding site. A positive aspect of conventional docking is the possibility of easily distributing the calculation on dedicated grid or cluster. The receptor is usually kept rigid, therefore the changes in the binding pocket geometry induced by the ligand is overlooked. Here we present a new docking approach (DynDock) that exploits molecular dynamics to preserve the flexibility of the receptor. To maintain high computational efficiency, DynDock has been developed to be distributed on a grid. The main advantages of this method are the full flexible molecular docking achieved during the simulation and the reduced number of compounds collected.

Keywords: Docking · Drug design · Molecular dynamics

1 Introduction

The molecular design is a computationally demanding task; it is the process of finding new drugs and involves the design of molecules that are complementary to the target in shape and charge. Usually, these compounds interact with a protein activating or inhibiting its function. There are two major methods of molecular design. The first is the Ligand-Based Drug Design (LBDD) that uses the structural characteristics of all molecules that bind the target of interest, to derive a pharmacophore model [1]. The second method is the Structure-Based Drug Design (SBDD), which is based on knowledge of the three-dimensional structure of the target [2]. The aim is to predict the affinity and the selectivity of a drug candidate using the ligand and the target structure.

In details, SBDD is a cyclic process, which starts from a known target structure usually experimentally obtained by X-ray crystallography or NMR spectroscopy [3]. The knowledge of 3D structures permits to run *in silico* studies to identify potential ligands (Fig. 1).

Following the molecular modelling predictions, the most promising compounds can be synthesized and evaluated for their biological properties. Once synthesized and

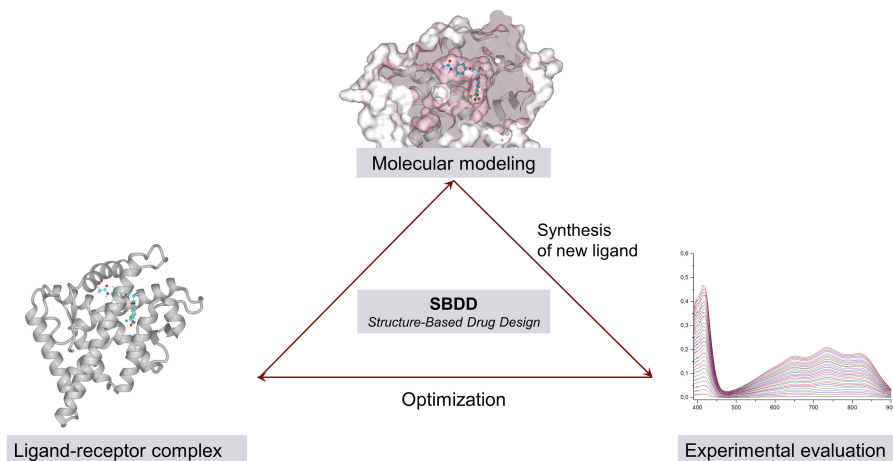


Fig. 1. Structure-based drug design cycle

tested, the new 3D structures can be solved and made ready for a further optimization cycle. This process reasonably permits to increase the affinity of new ligands but it is extremely costly.

Another limit of SBDD is that the experimental structures of the complex receptor/ligand are not always available and, when accessible, we must take into account that ligands can induce conformational changes in proteins and different ligands may stabilize different receptor conformations. Nevertheless, the crystallographic data represents only a successful binding event of a specific protein conformation and a specific ligand [4].

For these reasons, it is clear that the flexibility of the target receptor is an essential aspect that must be considered in the docking studies and it is not recommended to use only one structure of the receptor to perform the analysis.

Molecular dynamics (MD) is also a useful technique to evaluate critical phenomena and conformational changes involved in the molecular interactions [5, 6]. Keeping the proteins flexible in the molecular modelling studies has a high computational cost. The most popular docking programs limit the receptor flexibility to side-chain mobility only [7, 8]. In some other cases, they consider several snapshots extracted from a molecular dynamics of the receptor. This approach assumes that the protein flexibility could be encoded in an arbitrary set of MD conformations, but the molecular dynamics is strongly dependent on the ligand nature [4, 9].

The computational complexity of the procedure grows quickly with the numbers of atoms in the ligands. An exhaustive analysis of all possible ligands is far impossible even when a molecule is simplified in groups of atoms or residues. For example, the investigation of a very short peptide of only 10 amino acids, having as a starting point only 3 conformations (alpha, beta and coil), leads to more than $6 \cdot 10^7$ possible sequences, a number beyond the current computational possibilities.

DynDock employs an *in silico* combinatorial molecular dynamics to optimize ligands inside the target protein. This procedure combines the advantages of the SBDD

method with the accuracy of MD, reducing drastically the number of the possible sequences to analyse.

2 Methods

Figure 2 describes the DynDock approach.

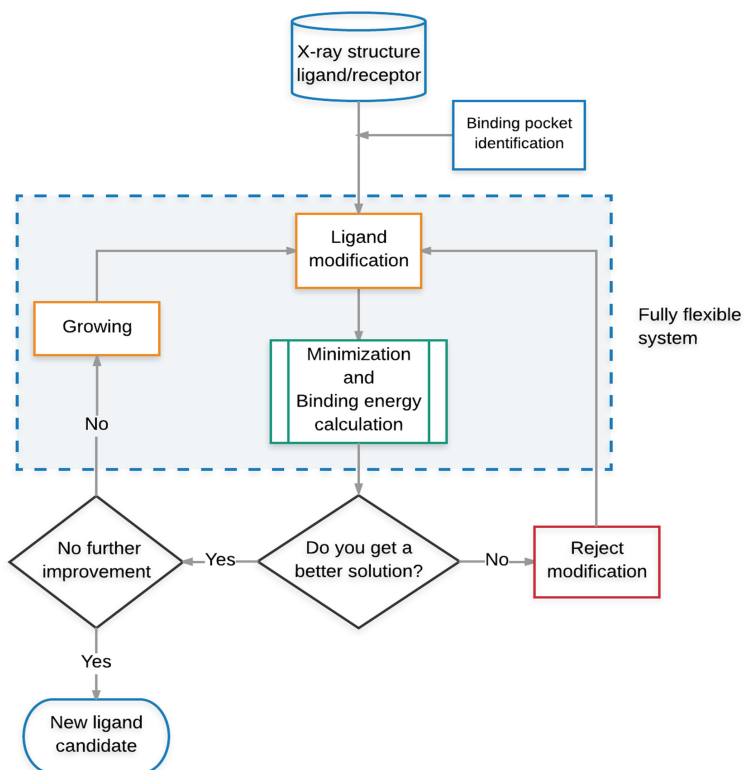


Fig. 2. DynDock flowchart

The DynDock is a hierarchical method to design a ligand candidate. We start from a fragment of a known ligand and we proceed with a cycle in which we evaluate the effect of a set of possible moves. The single move can be an addition, a deletion, an atom or residue swap, or a cyclization. After the move, we perform a molecular dynamics run of 1 ns and then we anneal the system. After the relaxation, we calculate the binding energy of the ligand and the distortion energy of the receptor. At the end of the cycle, the ligand having the highest binding energy is chosen for a further cycle. The procedure ends when no further energy improvement is observed. DynDock ensures to find always a better ligand, though it cannot guarantee to find the best one.

2.1 Fragment-Based Molecular Dynamics

The starting point for DynDock is the structure of a target protein bound to a ligand. In this work, we used as initial structures the microbial enzyme *Streptomyces griseus* Protease A (SGPA) [10] in complex with the peptide Pro-Ala-Pro-Tyr (5SGA PDB). It is a proteolytic enzyme with a serine residue (Ser 195) in its active site. The proximity of a histidine and an aspartate is essential for its activity (see the ligand diagram interaction in Fig. 2a) [11, 12].

The first step of DynDock is the preparation of the receptor structure for the docking, removing all water molecules and adding missing hydrogens. From the experimental structure of the ligand/receptor complex, it is possible to identify the binding site in which the ligand is grown. From literature data, we know that the active site of SGPA is an external portion of the receptor and the residue of tyrosine of the ligand interacts with the catalytic triad. In Fig. 2b, it is possible to see the position of the ligand PAPY on the external surface of the enzyme. The surface is colored by atom types. To put in evidence the active site, we colored only the residue with distance 4 Å from the ligand (Fig. 3).

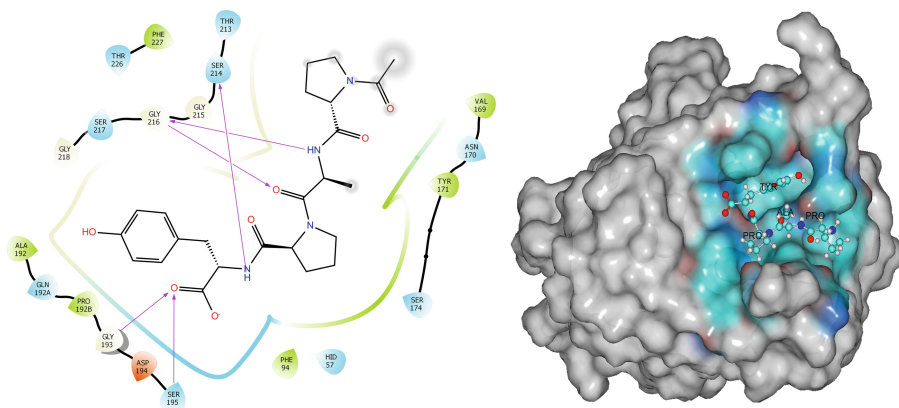


Fig. 3. (a) 2D interaction between the ligand and the enzyme. (b) Molecular surface of Serine protease in complex with PAPY inhibitor (5SGA PDB). (Color figure online)

The tyrosine residue is critical for the enzyme inhibition, and therefore, we have chosen the terminal Tyr as starting point for peptide elongation.

2.2 Preliminary Screening

The elongation procedure consists in adding an amino acid chosen among the 20 natural residues. Each amino acid, when binds a peptide, forms an amide bond characterized by two torsion angles. They describe the rotations of the polypeptide backbone around the bonds between N-C α (φ) and C α -C (ψ). It is well known that three regions of φ , ψ correspond to the most stable conformations namely α -helices, β -sheet

and coil. Therefore, for each residue added by DynDock, we must take into account the different geometries, and the number of possible structures becomes:

$$peptides = (conf * aa)^{elongation} \quad (1)$$

Where *conf* is the number of conformational regions, and *aa* is the number of amino acids. That for 3 geometries only and for an elongation of 10 residues, gives:

$$peptides = (3 \cdot 20)^{10} \cong 6.05 \cdot 10^{17} \quad (2)$$

Though the molecular dynamics approach promises a big progress in the docking field, it is evident that the astronomic number of possible peptides renders this way unmanageable. The DynDock approach greatly reduces the number of possible peptides and only the best *n*-mers can be kept for further elongation. If only the peptide with the highest binding energy is kept for each iteration, the number of peptides to analyze may collapse to:

$$peptides = elongation \cdot conf \cdot aa = 10 \cdot 3 \cdot 20 = 600 \quad (3)$$

Unfortunately, an important aspect of peptide folding was neglected in the above consideration. The interaction of a peptide with a receptor is not a simple cumulative process because the peptide residues can interact with the peptide itself changing conformation and then altering the binding with the receptor. Consequently, we cannot keep the best residue only for each elongation step, but we can safely choose to keep the a few number of peptides (*bestResults*) for each step. The number of possible peptides for *BestResults* = 5 becomes:

$$peptides = bestResults \cdot elongation \cdot conf \cdot aa = 5 \cdot 10 \cdot 3 \cdot 20 = 3000 \quad (4)$$

Considering the last 2 or 3 residues (*nmer*) during an elongation should give more realistic results, but the number of possible peptides would reach soon extremely large numbers (see Table 1) according to Eq. (5).

$$peptides = bestResults \cdot elongation \cdot (conf \cdot aa)^{nmer} \quad (5)$$

We have faced this problem in two different ways. The first is the distribution of the calculation on a dedicated grid. We have used GRIMD [13], an info structure that permits easily the delivery of molecular dynamics calculation on available PCs. The second solution is more chemistry-oriented. A preliminary screening is made before the

Table 1. Number of sequences to analyze based on the number of residues kept in memory

Elongation	Sequences to analyze
1 amino acid	3000
2 amino acids	180000
3 amino acids	$1.08 \cdot 10^7$

ligand fragment building. This step is essential to reduce the time of the entire drug design. The user can set the elongation (e.g. 6 amino acids) and the solubility of the ligand candidate. The solubility is an important parameter to be considered, in order to ensure the possibility of chemical synthesis and the biological screening. Water solubility can be predicted as a function of the surface hydrophobicity of the ligand. The tendency of a protein to aggregate and so to decrease its water solubility can be related to the hydrophobic surface [14]. Ligand candidates with potential low water solubility are not considered.

Finally, in order to avoid too exotic peptides, we introduced a phylogenetic control of sequences. We downloaded the human proteome from the UniProt database [15] (Proteome ID UP000005640) and we calculated the dimer abundance. We assigned a different weight based on dimer probabilities.

The DynDock method favors the building of dimers with high frequency.

2.3 Ligand Fragment Modification

The ligand fragment modification starts considering a swap, deletion and addition of amino acid residues.

The ligand modification step occurs during molecular dynamics simulations (MD), which means the ligand fragment and the receptor, are always in contact. A cubic cell of $57 \times 57 \times 57 \text{ \AA}$ was built around all atoms under periodic boundary condition. The MD simulation is set at 298 K with 1.25 fs of integration time steps for intramolecular forces. After each modification, the system is left to move for 1 ns to allow the receptor to better accommodate the ligand. The all-atom structure of the complex fragment/target is minimized using the force field AMBER14 [16] and the steepest descent minimization followed by a simulated annealing minimization [17].

2.4 Output Selection

The binding affinity of the ligand was calculated using the function *YaEnergy* already reported in [18]. *YaEnergy* permits to estimate the binding energy taking into account the biological history of the receptor. It has been written after an extensive genetic algorithm including a term that depends on the minimal distance between the ligand barycenter and the nearest conserved residues. The sequence of the enzyme SGPA is highly conserved through species indicating that the sequence has been maintained by evolution despite speciation. As shown in Fig. 4, the residues of the binding pocket are extremely conserved confirming that functional residues are generally more preserved [19]. The conservation string was obtained from the ConSurf database [20], a server for identification of structurally important residues in protein sequences (<http://conseq.tau.ac.il/>).

The binding energy calculated at the end of the molecular dynamics was used to build up a new ranking function for peptide selection. Whereas the choice of high binding energies is straightforward, energy alone is not enough because it tends to bias longer peptides. A long peptide, in fact, can interact in more ways than a shorter one. The rank function at denominator has the peptide length and a negative surface area term at numerator. There is also a corrective term based on the receptor distortion.

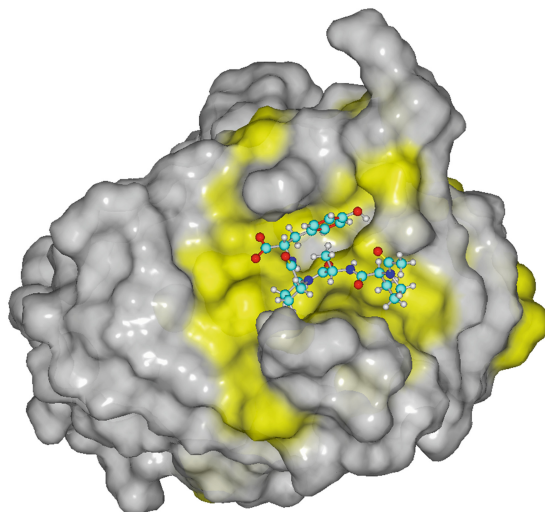


Fig. 4. 5SGA surface. The conserved residues are colour mapped in yellow onto the protein surface. (Color figure online)

The rationale is that a peptide, modifying the 3D structure of the flexible receptor, can drive its geometry far from the experimental data. In the ranking function, we have added a term to award receptor structures not dissimilar from the crystallographic data.

$$\text{DynDock rank} = \frac{yaEnergy + 20 \cdot \frac{En.end}{En.in} - \frac{SurfArea}{90}}{Length} \quad (6)$$

Whereas $YaEnergy$ is the peptide binding energy, $En.in$ and $En.end$ are the initial and final energies of the receptor calculated with the force field AMBER14, and $SurfArea$ is the molecular surface area of the peptide.

DynDock selects the molecules with the highest binding energy and lowest deformation of the receptor. The receptor changes its structure during the fragment growing to better accommodate the ligand. This could damage the 3D structure and lead to an unrealistic structure. For this reason, the DynDock rank function is rescaled on the dimension of the ligands, optimizing the binding energy value with the ligand surface area. This is essential to prevent that the algorithm prefers bigger peptides that, having more atoms, have more chances to interact with the receptor.

Based on the DynDock rank value, the step that involves the ligand modification can be accepted or rejected.

The process ends when there are no further energy improvements. The computational complexity of the procedure grows quickly with the numbers of conformers considered. Consequently, to reduce the computational time and cost we have used a specialized grid (GRIMD) to distribute the calculation [13].

3 Results and Discussion

The evaluation of the DynDock procedure can be done in terms of binding energy. Interestingly, following the methods herein described, after binding each peptide can be forced to leave the receptor and the activation energy required to leave the receptor estimated. This calculation permits to evaluate the residence time [21] of the ligand (that is the inverse of the unbinding kinetic constant rate k_{off}) [22–24] that is of fundamental importance in drug discovery.

Starting from the first amino acid (tyrosine), DynDock adds new amino acids and chooses the best dimer among the 20 possibilities. The dimer was then further elongated until the length of 4 residues. All ligands are ordered by length and by ranking value. Based on the rank function value, DynDock selects only two dimers made, in this example, glutamine-tyrosine (QY) and alanine-tyrosine (AY) and proceeds further with the elongation. We decided to fix the elongation to 4 residues to make easier the comparison with the crystallographic structure. The best ligand developed by DynDock protocol (PGAY) shows higher binding affinity than the experimental molecule PAPY and it still maintains the interactions with the catalytic triad (see Fig. 5).

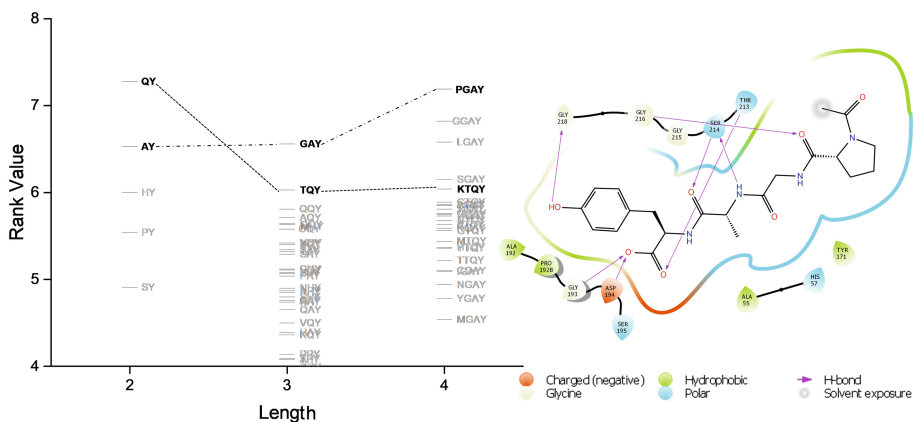


Fig. 5. DynDock protocol trend. Length is the number of amino acids in the ligand and the rank function is the normalized binding energy. On the left are shown the ligand PGAY interactions with the binding site.

To build a ligand candidate formed by ten residues, traditional approaches of structure-based drug design provide 10 million of millions of sequences to analyze ($20^{10} = 1 \cdot 10^{13}$). DynDock method limits the calculation to a number of sequences to analyze of few thousands.

4 Conclusion

The method here described for Drug Design is an easy way to perform fully molecular docking reducing drastically the number of possible sequences. It permits also the easy distribution on computer grids to further reduce the analysis time.

This hierarchical approach has several advantages over traditional docking. First, the flexibility of the receptor, essential for its function, is fully considered and modeled with the modern AMBER14 force field. Second, DynDock takes into account the receptor distortion to avoid unrealistic and improbable interactions. Third, the sequential procedure guarantees to find a series of peptides with high binding energies without a sensible decrease of computational performances. Fourth, the sequentiality of the investigation makes DynDock ideal for parallelization or for use on grids. Finally, the molecular dynamics can be used also to perform a steered molecular dynamics of the ligand out from the receptor to estimate the residence time. This improvement will be the object of an upcoming paper.

References

1. Sliwoski, G., Kothiwale, S., Meiler, J., Lowe, E.W.: Computational methods in drug discovery. *Pharmacol. Rev.* **66**, 334–395 (2014)
2. Salum, L.B., Polikarpov, I., Andricopulo, A.D.: Structure-based approach for the study of estrogen receptor binding affinity and subtype selectivity. *J. Chem. Inf. Model.* **48**, 2243–2253 (2008)
3. Ferreira, L.G., Dos Santos, R.N., Oliva, G., Andricopulo, A.D.: Molecular docking and structure-based drug design strategies. *Molecules* **20**, 13384–13421 (2015)
4. Sessa, L., Biasi, L.D., Concilio, S., Cattaneo, G., De Santis, A., Iannelli, P., Piotto, S.: A new flexible protocol for docking studies. *Commun. Comput. Inf. Sci.* **587**, 117–126 (2016)
5. Lin, J.-H.: Accommodating protein flexibility for structure-based drug design. *Curr. Top. Med. Chem.* **11**, 171–178 (2011)
6. Durrant, J.D., McCammon, J.A.: Molecular dynamics simulations and drug discovery. *BMC Biol.* **9**, 71 (2011)
7. de Ruyck, J., Brysbaert, G., Blossey, R., Lensink, M.F.: Molecular docking as a popular tool in drug design, an in silico travel. *Adv. Appl. Bioinf. Chem. AABC* **9**, 1 (2016)
8. Geng, C., Narasimhan, S., Rodrigues, J.P., Bonvin, A.M.: Information-driven, ensemble flexible peptide docking using HADDOCK. In: Schueler-Furman, O., London, N. (eds.) *Modeling Peptide-Protein Interactions. MMB*, vol. 1561, pp. 109–138. Springer, New York (2017). https://doi.org/10.1007/978-1-4939-6798-8_8
9. Sessa, L., Concilio, S., Piotto, S.: Molecular dynamics and morphing protocols for high accuracy molecular docking. In: Piotto, S., Rossi, F., Concilio, S., Reverchon, E., Cattaneo, G. (eds.) *Advances in Bionanomaterials. LNB*, pp. 85–96. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-62027-5_8
10. James, M.N.G., Sielecki, A.R., Brayer, G.D., Delbaere, L.T.J., Bauer, C.A.: Structures of product and inhibitor complexes of *Streptomyces griseus* protease A at 1.8 Å resolution: a model for serine protease catalysis. *J. Mol. Biol.* **144**, 43–88 (1980)
11. Bartholomae, M., Buivydas, A., Viel, J.H., Montalban-Lopez, M., Kuipers, O.P.: Major gene-regulatory mechanisms operating in ribosomally synthesized and post-translationally modified peptide (RiPP) biosynthesis. *Mol. Microbiol.* **106**(2), 186–206 (2017)

12. Harish, B., Uppuluri, K.B.: Microbial serine protease inhibitors and their therapeutic applications. *International J. Biol. Macromol.* **107**, 1373–1387 (2017)
13. Piotto, S., Di Biasi, L., Concilio, S., Castiglione, A., Cattaneo, G.: GRIMD: distributed computing for chemists and biologists. *Bioinformatics* **10**, 43–47 (2014)
14. Wagner, J.R., Sorgentini, D.A., Añón, M.C.: Relation between solubility and surface hydrophobicity as an indicator of modifications during preparation processes of commercial and laboratory-prepared soy protein isolates. *J. Agric. Food Chem.* **48**, 3159–3165 (2000)
15. UniProt Consortium: UniProt: the universal protein knowledgebase. *Nucleic Acids Res.* **45**, D158–D169 (2016)
16. Lee, J., Cheng, X., Swails, J.M., Yeom, M.S., Eastman, P.K., Lemkul, J.A., Wei, S., Buckner, J., Jeong, J.C., Qi, Y.: CHARMM-GUI input generator for NAMD, GROMACS, AMBER, OpenMM, and CHARMM/OpenMM simulations using the CHARMM36 additive force field. *J. Chem. Theory Comput.* **12**, 405–413 (2015)
17. Krieger, E., Vriend, G.: YASARA view-molecular graphics for all devices-from smartphones to workstations. *Bioinformatics* **30**, 2981–2982 (2014)
18. Di Biasi, L., Fino, R., Parisi, R., Sessa, L., Cattaneo, G., De Santis, A., Iannelli, P., Piotto, S.: Novel algorithm for efficient distribution of molecular docking calculations. *Commun. Comput. Inf. Sci.* **587**, 65–74 (2016)
19. Piotto, S., Di Biasi, L., Fino, R., Parisi, R., Sessa, L., Concilio, S.: Yada: a novel tool for molecular docking calculations. *J. Comput. Aided Mol. Des.* **30**, 753–759 (2016)
20. Berezin, C., Glaser, F., Rosenberg, J., Paz, I., Pupko, T., Fariselli, P., Casadio, R., Ben-Tal, N.: ConSeq: the identification of functionally and structurally important residues in protein sequences. *Bioinformatics* **20**, 1322–1324 (2004)
21. Copeland, R.A., Pompliano, D.L., Meek, T.D.: Drug–target residence time and its implications for lead optimization. *Nat. Rev. Drug Discovery* **5**, 730–739 (2006)
22. Piotto, S., Sessa, L., Iannelli, P., Concilio, S.: Computational study on human sphingomyelin synthase 1 (hSMS1). *Biochim. Biophys. Acta (BBA) Biomembr.* **1859**, 1517–1525 (2017)
23. Casas, J., Iburguren, M., Álvarez, R., Terés, S., Lladó, V., Piotto, S.P., Concilio, S., Busquets, X., López, D.J., Escribá, P.V.: G protein-membrane interactions II: effect of G protein-linked lipids on membrane structure and G protein-membrane interactions. *Biochim. Biophys. Acta (BBA) Biomembr.* **1859**, 1526–1535 (2017)
24. Piotto, S., Trapani, A., Bianchino, E., Iburguren, M., López, D.J., Busquets, X., Concilio, S.: The effect of hydroxylated fatty acid-containing phospholipids in the remodeling of lipid membranes. *Biochim. Biophys. Acta (BBA) Biomembr.* **1838**, 1509–1517 (2014)



Stochastic Numerical Models of Oscillatory Phenomena

Raffaele D'Ambrosio¹(✉), Martina Moccaldi², Beatrice Paternoster²,
and Federico Rossi³

¹ Department of Engineering and Computer Science and Mathematics,
University of L'Aquila, L'Aquila, Italy
raffaele.dambrosio@univaq.it

² Department of Mathematics, University of Salerno, Fisciano, Italy
{mmoccaldi, beapat}@unisa.it

³ Department of Chemistry and Biology, University of Salerno, Fisciano, Italy
frossi@unisa.it

Abstract. The use of time series for integrating ordinary differential equations to model oscillatory chemical phenomena has shown benefits in terms of accuracy and stability. In this work, we suggest to adapt also the model in order to improve the matching of the numerical solution with the time series of experimental data. The resulting model is a system of stochastic differential equations. The stochastic nature depends on physical considerations and the noise relies on an arbitrary function which is empirically chosen. The integration is carried out through stochastic methods which integrate the deterministic part by using one-step methods and approximate the stochastic term by employing Monte Carlo simulations. Some numerical experiments will be provided to show the effectiveness of this approach.

Keywords: Oscillating solutions · Belousov-Zhabotinsky reaction
Reaction equations · Stochastic chemical oscillators
Stochastic models · Stochastic differential equations

1 Introduction

This work deals with the problem of modelling oscillatory phenomena by suitable systems of differential equations, together with providing a proper numerical scheme for an accurate and efficient approximation of their solutions. A special emphasis is here given to a significant case study: the well-known Belousov-Zhabotinsky (BZ) reaction. The BZ is a striking example of a self-organizing chemical system, and thanks to its characteristics, it became a widely employed model also in other fields. For instance, in biology BZ can be considered a simple analogue of periodic phenomena (metabolic cycles, circadian clocks, etc) and in mathematics and physics it is an ideal example of complex nonlinear dynamical system [1]. There are several models to describe the complex kinetics of the BZ reaction, being the *Oregonator*, the most used [1–3].

In [4] this system has been integrated by employing an adapted numerical scheme which exploits information obtained by observing time series of experimental data. It has been shown that this problem-oriented approach is more accurate and stabler than general-purpose numerical methods, which could require a strong reduction in stepsize in order to accurately follow the behaviour of the exact solution. In this paper, we focus on the nature of the operator, in order to improve the matching of the numerical solution of the Oregonator with the time series.

In Field, Körös and Noyes approach, the time evolution of BZ reactions is treated as a continuous and deterministic process. In many cases, this is sufficient to study the qualitative behaviour of the system. However, the reaction-rate equations may be unable to describe the fluctuations in the molecular population levels within the study, for instance, of ecological systems, microscopic biological systems and nonlinear systems characterised by chemical instability. Therefore, in some cases it may be more convenient to employ a stochastic approach, which derives from some physical considerations. Firstly, molecular population levels change in a discrete manner, so the time evolution of a chemically reacting system is not a continuous process. Moreover, it is impossible to predict the exact molecular population levels at a certain time unless the exact positions and velocities of all the molecules in the system are known [5,6]. The stochastic formulation of chemical kinetics basically takes into account that the collisions in a system of molecules in thermal equilibrium occur essentially in a random way. However, it is based on the so-called master equation which is often mathematically intractable. For this reason, we suggest to add a stochastic term to the deterministic system in order to obtain a model which is still simple to integrate like the reaction-rate equations, but it can lead to a numerical solution more similar to the time series. In the resulting model, the time evolution of the system is described by a system of Itô stochastic differential equations, where the stochastic term is characterised by a Wiener process and an arbitrary function empirically chosen. The deterministic term of this system is integrated by employing a one-step numerical method, whereas the stochastic term is approximated through Monte Carlo simulations. The numerical solution is compared to the time series of the experiment performed in [7] on an unstirred ferroin catalysed BZ system.

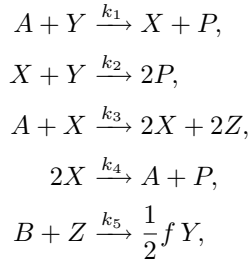
In summary, we describe the main aspects of Belousov-Zhabotinsky reaction in Sect. 2, Sect. 3 is devoted to the development of the new stochastic model to describe the kinetics of this reaction, while Sect. 4 shows some numerical experiments and Sect. 5 exhibits the conclusions.

2 The Belousov Zhabotinsky Reaction

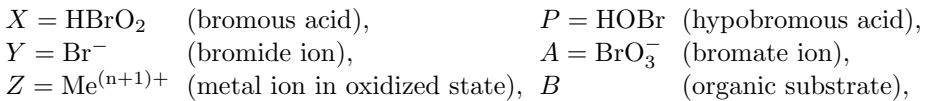
The Belousov-Zhabotinsky reaction is probably the simplest closed macroscopic system that can be maintained far from equilibrium by an internal source of free energy homogeneously distributed in space [8–11]. Being outside of thermodynamical equilibrium, BZ can display several *exotic* dynamical regimes: periodic,

aperiodic and chaotic oscillations [12–14], Turing structures and pattern formation [15,16], autocatalysis and bistability [17]. At present, most of the research involving the BZ reaction deals with stimuli-responsive smart materials [18–20] and with the simulation of complex biological communication [21–23]. In this work, we attempt to reproduce the periodic oscillatory regime generally manifested by the BZ in homogeneous well-stirred reactors.

BZ reaction involves an organic substrate that is oxidised by bromate ions in an acidic medium and is generally catalysed by one-electron metal-ion oxidants with standard reduction potentials of 1–1.5 V, for example metal ions complexes (ferroin, cerium sulphate, etc.) [1,3] (and references therein). Under proper conditions, the system exhibits self-sustained temporal oscillations in the concentrations of the catalysts, visible through a color change in the solution (more drastic for the iron). The oscillations stem from two concurrent processes: at the beginning the metal ion is reduced and the concentration of bromide ions ($[\text{Br}^-]$) is high (Process I); then bromides are consumed up to a certain critical value and the metal ion is oxidised (Process II); finally, the metal ion reacts to produce bromide ions and reverts to its reduced state again. However, from the kinetics point of view, the oscillations are caused by an Hopf instability deriving from the nonlinear chemical mechanism (autocatalysis + inhibition) and occurring in the reaction. The most widely accepted model to describe BZ reaction has been proposed by Field and Noyes in [24] and it has been derived from the more complicated Field-Körös-Noyes mechanism [25] which is based on 11 reactions involving 15 chemical species that lead to a system of 7 coupled nonlinear first-order ordinary differential equations. In order to theoretically analyse oscillations, bistability and traveling waves, it is sufficient to consider the following reduced formulation of the FKN mechanism [26]:



where the main chemical elements are:



and f is a stoichiometric factor which represents the number of bromide ions produced when metal ions are reduced. The concentrations of A , B and P are generally maintained constant, whereas the concentrations of intermediates X , Y and Z vary periodically. The kinetics of the system can be described by the following set of 3 differential equations [3]

$$\frac{dx^*}{dt^*} = k_1 a y^* - k_2 x^* y^* + k_3 a x^* - 2k_4 (x^*)^2, \quad (1a)$$

$$\frac{dy^*}{dt^*} = -k_1 a y^* - k_2 x^* y^* + \frac{f}{2} k_5 b z^*, \quad (1b)$$

$$\frac{dz^*}{dt^*} = 2k_3 a x^* - k_5 b z^*, \quad (1c)$$

which is called *Oregonator* and involves the concentrations of the aforementioned chemical elements. We refer to such concentrations by using letters in lower case henceforth. As highlighted in [27], the Oregonator is not only the simplest model for Belousov-Zhabotinsky reaction but also the most popular to study the period and the amplitude of observed oscillations.

The oscillations in the exact solution of the Oregonator are strongly dependent on the values of the involved parameters, especially k_5 and f . Indeed, if $k_5 = 0$, the bromide ion (Br^-) concentration decays to zero according to the Eq. (1b), so the system cannot oscillate. Moreover, oscillations occur only if $0.5 < f < 2.414$, whereas for $f < 0.5$ and $f > 2.414$ the system is in a stable steady state, being Process II or Process I dominant, respectively (see [1] and references therein).

In order to integrate the Oregonator (1), we consider its dimensionless form, as follows:

$$\epsilon \frac{dx}{dt} = qy - xy + x(1 - x), \quad (2a)$$

$$\epsilon' \frac{dy}{dt} = -qy - xy + fz, \quad (2b)$$

$$\frac{dz}{dt} = x - z, \quad (2c)$$

where

$$\begin{aligned} x &= \frac{2k_4}{k_3 a} x^*, & y &= \frac{k_2}{k_3 a} y^*, & z &= \frac{k_4 k_5 b}{(k_3 a)^2} z^*, & t &= t^* k_5 b, \\ \epsilon &= \frac{k_5 b}{k_3 a}, & \epsilon' &= \frac{2k_4 k_5 b}{k_2 k_3 a}, & q &= \frac{2k_1 k_4}{k_2 k_3}, \end{aligned} \quad (3)$$

or, in a more compact form,

$$\frac{dr}{dt} = F(r; q, f, \epsilon, \epsilon'), \quad (4)$$

where $r = [x, y, z]^T$ and $F(r; q, f, \epsilon, \epsilon') = \begin{bmatrix} \frac{1}{\epsilon} (qy - xy + x(1 - x)) \\ \frac{1}{\epsilon'} (-qy - xy + fz) \\ x - z \end{bmatrix}$.

3 Stochastic Adaptation of the Oregonator

We aim to develop a simple stochastic variant of the deterministic system (4) in order to better describe the fluctuations usually observed in time series of

experimental data. For this purpose, we add a stochastic term to the reaction-rate Eq. (4), as follows

$$\frac{dR}{dt} = F(R; q, f, \epsilon, \epsilon') + \lambda G(R) dW, \quad (5)$$

where $R(t)$ is a three-dimensional stochastic process describing the concentrations of the key chemical elements, $F(R; q, f, \epsilon, \epsilon')$ is the deterministic forcing term, λ is the amplitude of the stochastic term, $G(R)$ is an arbitrary function and $W(t)$ is a Wiener process. We recall that a standard Wiener process is a stochastic process $\{W(t), t \in [0, T]\}$ such that $W(0) = 0$ with probability 1, the function $\Phi : t \rightarrow W(t)$ is continuous with probability 1, the increments are independent and behave as the random variable $\sqrt{t-s} \mathcal{N}(0, 1)$, i.e. a normally distributed random variable with zero-mean and variance equal to $t-s$.

Equation (5) is a system of Itô stochastic differential equations, whose solution $R(t)$ is a stochastic process depending on an initial value

$$R(0) = R_0, \quad (6)$$

a deterministic integral and an Itô stochastic integral.

In order to integrate (5) in $[0, T]$, we discretize the interval by selecting equidistant $N+1$ points, as follows

$$0 = t_0 < t_1 < \dots < t_N = T,$$

and we employ a one-step stochastic numerical method having this general formulation

$$R_{n+1} = R_n + h(\alpha F(R_{n+1}) + \beta F(R_n)) + G(R_n) (W(t_{n+1}) - W(t_n)), \quad (7)$$

where h is the integration stepsize. For the simulation of the Wiener increments, we employ Monte Carlo simulations, i.e. we generate a standard normally distributed variable through the Matlab routine `randn` and we approximate the Wiener increments multiplying this variable with \sqrt{h} .

4 Numerical Experiments

We take into account the experiment performed in [7] on an unstirred ferroin catalysed BZ system, where the organic substrate is the malonic acid ($B = \text{MA}$) and the catalyst is the redox couple ferriin/ferroin ($\text{Fe}(\text{phen})_3^{3+}/\text{Fe}(\text{phen})_3^{2+}$).

In [7] time series are recorded spectrophotometrically at wavelengths equal to 510 nm (ferroin) and 630 nm (ferriin), where the molar extinction coefficients are equal to $1.1 \times 10^4 \text{ mol}^{-1} \text{ dm}^3 \text{ cm}^{-1}$ and $620 \text{ mol}^{-1} \text{ dm}^3 \text{ cm}^{-1}$, respectively. Employing these data, we construct the corresponding time series of the concentration of the ferriin, i.e. the catalyst in its oxidized state, which is the third component of the solution of the Oregonator (5). The resulting time series shows an initial exponential decay trend corresponding to the start of the reaction (see Fig. 1) and followed by periodic oscillations.

In order to model this chemically reacting system, we consider the system of Itô stochastic differential equations (5) in a region of the plane $k_5 - f$ where the solution is known to oscillate. With regards to the choice of $G(R)$, we select different functions. Firstly, we have considered a linear noise depending on the parameters of the problem

$$G_{\text{lin}}(x, y, z) = \left[x + \frac{q}{\varepsilon}y + 1, y + \frac{q}{\varepsilon}y, z + \frac{q}{\varepsilon}y \right]^{\top}. \quad (8)$$

This choice is convenient because the function evaluations are not highly demanding in terms of computational cost. Another possible G -function has a logarithmic expression:

$$G_{\text{log}}(x, y, z) = \left[2 \log(xy), \log\left(\frac{y^2}{q}\right), \log(z^2 + 1) \right]^{\top}. \quad (9)$$

Since we observe an oscillatory behaviour in time series (Fig. 1), we next consider a simple trigonometric noise

$$G_{\text{trig}}(x, y, z) = [\sin(x), \sin(y), \sin(z)]^{\top}, \quad (10)$$

but, as will be shown in Table 1, it may be more convenient to adopt a trigonometric G -function depending on the parameters of the problem, as follows:

$$G_{\text{pdtrig}}(x, y, z) = \left[\sin(x) + \frac{q}{\varepsilon} \sin(y) + 1, \sin(y) + \frac{q}{\varepsilon'} \sin(y), \sin(z) + \sin(y) \right]^{\top}. \quad (11)$$

We have solved system (5) in $[0, 250]$ combined with these different noises, provided by the initial conditions

$$x(0) = 0.0013, \quad y(0) = 0.2834, \quad z(0) = 0.1984, \quad (12)$$

and the following values for the parameters

$$f = 1, \quad q = 3.52 \cdot 10^{-5}, \quad \varepsilon = 0.3779, \quad \varepsilon' = 7.56 \cdot 10^{-4}. \quad (13)$$

We remark that the concentrations in (12) are in their dimensionless form. We employ a one-step method to integrate the deterministic part and Monte Carlo simulations to treat the stochastic term. In particular, we have integrated the deterministic term through explicit Euler method, obtaining the Euler-Maruyama method

$$R_{n+1} = R_n + h F(R_n) + G(R_n) (W(t_{n+1}) - W(t_n)). \quad (14)$$

However, this method is strongly unstable for every choice of the G functions and amplitude λ due to the stiffness of the problem. For this reason, we integrate the deterministic term through the implicit trapezoidal rule, as follows:

$$R_{n+1} = R_n + \frac{h}{2} (F(R_n) + F(R_{n+1})) + G(R_n) (W(t_{n+1}) - W(t_n)). \quad (15)$$

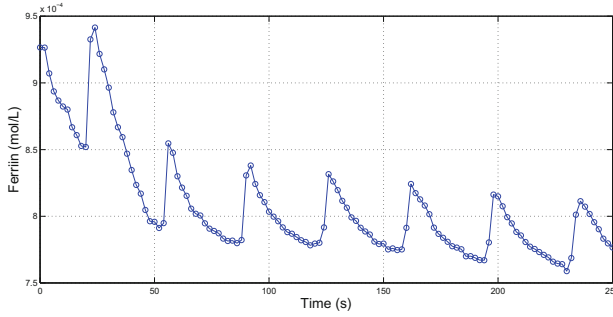


Fig. 1. Time series of concentration of ferriin related to the experiment carried out in [7] on an unstirred ferriin catalyzed BZ system.

Table 1. Minimum error computed in the last point with 100 simulations as difference between the value assumed in the last point by the numerical solution obtained by the scheme (15) for $h = 0.06$ and with different functions G and the corresponding value observed in time series.

λ	G_{lin}	G_{log}	G_{trig}	G_{pdtrig}
0	$9.99 \cdot 10^{-1}$			
0.5	$1.24 \cdot 10^{-2}$	$8.99 \cdot 10^{-4}$	$1.45 \cdot 10^{-2}$	$6.56 \cdot 10^{-3}$
1	$1.25 \cdot 10^{-1}$	$1.29 \cdot 10^{-3}$	$2.84 \cdot 10^{-1}$	$1.92 \cdot 10^{-2}$

We remark that the implicitness of this stochastic method is only in the deterministic part.

Table 1 reports the relative errors computed by comparing the value assumed by the numerical solution in the last point of the interval and the corresponding value observed in time series. In case of non-zero amplitude of the stochastic term, i.e. when the system (5) does not reduce to the deterministic formulation (4), we have run 100 simulations for each G function and amplitude λ because of the random nature of the Wiener increments and we have computed the minimum error obtained. We observe that the accuracy generally improves when we add the stochastic term to the model and it is higher for the logarithmic noise (9) and the parameter-dependent trigonometric one (11) than for the linear (8) and the first trigonometric case (10). Moreover, increasing the amplitude λ of the stochastic term, the errors related to the stochastic models become higher, but they are still smaller than the error obtained with the deterministic formulation of the problem. Therefore, it may be convenient to add a stochastic term, but its amplitude has to be small enough, so that the noise does not cover the solution.

The solution of the deterministic model (see Fig. 2(a)) has a regular profile having only two oscillations, so it differs more from the time series than the solutions of stochastic models. Indeed, the numerical solution of the stochastic model combined with a trigonometric noise (see Fig. 2(d)) has a regular profile with three oscillations, but it is still far from time series. The choices of a

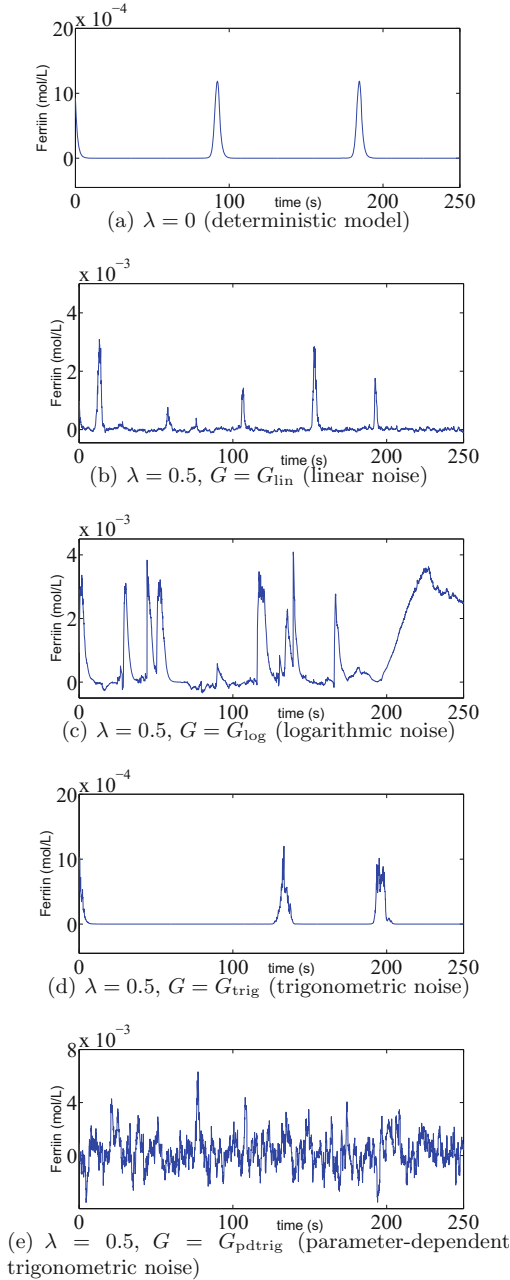


Fig. 2. Numerical solution of stochastic Oregonator (5) having initial conditions (12) and parameters (13) computed by the method (15) which integrates the deterministic term with the implicit trapezoidal rule and treats the stochastic part through Monte Carlo simulations. Different choices for the amplitude λ and the function G are considered and the adopted stepsize is $h = 0.06$.

linear noise (Fig. 2(b)) or a logarithmic noise (Fig. 2(c)) lead to solutions which are more oscillatory, so they are qualitatively more similar to time series. However, the solution obtained with a linear noise exhibits some spurious oscillations due to the noise and that one computed with a logarithmic noise has a highly irregular profile. As it regards Fig. 2(e), we can observe that the profile of the solution quantitatively matches very well with the time series and, moreover, the peaks are distributed similarly as in the pattern of the time series, which makes this kind of choice of the diffusion term in the stochastic model very promising. Clearly, the noisy behaviour observable in Fig. 2(e) is given by a single realization of the stochastic process solution that may be replaced, in future investigations, by a more regular and smooth mean behaviour over several realizations. We remark that in the figures the variable concentrations of ferriin (z) and time (t) have been recasted according to the positions (3) by employing the values

$$k_2 = 1.11 \cdot 10^6, \quad k_3 = 15.54, \quad k_4 = 1.11 \cdot 10^3, \quad k_5 = 1.$$

5 Conclusion

In this work, we have presented a new stochastic model to describe the kinetics of Belousov-Zhabotinsky reaction, assumed here as an experimental benchmark for proposing an adapted numerical scheme for differential models of oscillatory phenomena. Indeed, following the idea of adapting numerical schemes to time series presented in [4] (coming from [28–34] and references therein), we have adapted in this work also the model to describe better the available experimental data. In particular, we have considered the well-known deterministic model developed by Fields, Körös and Noyes and we have added a stochastic term, leading to a system of Itô stochastic differential equations. In this system, the stochastic term is characterized by an arbitrary function selected empirically. The resulting system has been integrated by a combination of known time-stepping methods for the integration of the deterministic part and Monte Carlo simulations for the numerical treatment of the stochastic term. The numerical solution has been compared with the time series related to the experiment carried out in [7] on an unstirred ferroin-catalysed BZ system. Numerical experiments show an high improvement in accuracy and a slight enhancement in the preservation of the qualitative behaviour observed in time series. It is important to highlight that our proposed approach can be assumed as a general setting for handling oscillatory problems in many different contexts: for instance, in the description of chemical oscillators in compartmentalized systems like microemulsions that feature nano-sized reactors [35]. Future developments of this research will be focused on taking these preliminary results as starting point to also fit the data into the model under a qualitative point of view, rather than only quantitative. In this sense, as it is clearly visible in the experiments, the passage from deterministic to stochastic models has been crucial and it seems promising to proceed in this direction.

References


1. Tyson, J.J.: What everyone should know about the Belousov-Zhabotinsky reaction. In: Levin, S.A. (ed.) *Frontiers in Mathematical Biology*. LNMB, vol. 100, pp. 569–587. Springer, Heidelberg (1994). https://doi.org/10.1007/978-3-642-50124-1_33
2. Epstein, I.R., Pojman, J.A.: *An Introduction to Nonlinear Chemical Dynamics: Oscillations, Waves, Patterns, and Chaos*, 1st edn. Oxford University Press, Oxford (1998)
3. Murray, J.D.: *Mathematical Biology*. Springer, New York (2004)
4. D'Ambrosio, R., Moccaldi, M., Paternoster, B., Rossi, F.: On the employ of time series in the numerical treatment of differential equations modeling oscillatory phenomena. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016*. CCIS, vol. 708, pp. 179–187. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_16
5. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **81**(25), 2340–2361 (1977)
6. Gillespie, D.T., Hellander, A., Petzold, L.R.: Perspective: stochastic algorithms for chemical kinetics. *J. Chem. Phys.* **138**, 170901 (2013)
7. Rossi, F., Budroni, M.A., Marchettini, N., Cutietta, L., Rustici, M., Liveri, M.L.T.: Chaotic dynamics in an unstirred ferroin catalyzed Belousov-Zhabotinsky reaction. *Chem. Phys. Lett.* **480**(4), 322–326 (2009)
8. Belousov, B.P.: An oscillating reaction and its mechanism. *Sborn. Referat. Radiat. Med. (Collection of Abstracts on Radiation Medicine)*, *Medgiz* **145** (1959)
9. Field, R.J., Burger, M.: *Oscillations and Traveling Waves in Chemical Systems*. Wiley-Interscience, New York (1985)
10. Zhabotinsky, A.M.: Periodic processes of the oxidation of malonic acid in solution (study of the kinetics of Belousov's reaction). *Biofizika* **9**, 306–311 (1964)
11. Zhabotinsky, A.M., Rossi, F.: A brief tale on how chemical oscillations became popular an interview with Anatol Zhabotinsky. *Int. J. Des. Nat. Ecodyn.* **1**(4), 323–326 (2006)
12. Sciascia, L., Rossi, F., Sbriziolo, C., Liveri, M.L.T., Varsalona, R.: Oscillatory dynamics of the Belousov-Zhabotinsky system in the presence of a self-assembling nonionic polymer. Role of the reactants concentration. *Phys. Chem. Chem. Phys.* **12**(37), 11674–11682 (2010)
13. Marchettini, N., Budroni, M.A., Rossi, F., Masia, M., Liveri, M.L.T., Rustici, M.: Role of the reagents consumption in the chaotic dynamics of the Belousov-Zhabotinsky oscillator in closed unstirred reactors. *Phys. Chem. Chem. Phys.* **12**(36), 11062–11069 (2010)
14. Rossi, F., Budroni, M.A., Marchettini, N., Carballido-Landeira, J.: Segmented waves in a reaction-diffusion-convection system. *Chaos Interdisc. J. Nonlinear Sci.* **22**(3), 037109 (2012)
15. Budroni, M.A., Rossi, F.: A novel mechanism for in situ nucleation of spirals controlled by the interplay between phase fronts and reaction-diffusion waves in an oscillatory medium. *J. Phys. Chem. C* **119**(17), 9411–9417 (2015)
16. Rossi, F., Ristori, S., Rustici, M., Marchettini, N., Tiezzi, E.: Dynamics of pattern formation in biomimetic systems. *J. Theor. Biol.* **255**(4), 404–412 (2008)
17. Taylor, A.F.: Mechanism and phenomenology of an oscillating chemical reaction. *Prog. React. Kinet. Mech.* **27**(4), 247–325 (2002)
18. Souza, T.P., Perez-Mercader, J.: Entrapment in giant polymersomes of an inorganic oscillatory chemical reaction and resulting chemo-mechanical coupling. *Chem. Commun.* **50**(64), 8970–8973 (2014)

19. Tamate, R., Ueki, T., Shibayama, M., Yoshida, R.: Self-oscillating vesicles: spontaneous cyclic structural changes of synthetic diblock copolymers. *Angew. Chem. Int. Ed.* **53**(42), 11248–11252 (2014)
20. Epstein, I.R., Xu, B.: Reaction-diffusion processes at the nano- and microscales. *Nat. Nanotechnol.* **11**(4), 312–319 (2016)
21. Torbensen, K., Rossi, F., Pantani, O.L., Ristori, S., Abou-Hassan, A.: Interaction of the Belousov-Zhabotinsky reaction with phospholipid engineered membranes. *J. Phys. Chem. B* **119**(32), 10224–10230 (2015)
22. Torbensen, K., Rossi, F., Ristori, S., Abou-Hassan, A.: Chemical communication and dynamics of droplet emulsions in networks of Belousov-Zhabotinsky micro-oscillators produced by microfluidics. *Lab Chip* **17**(7), 1179–1189 (2017)
23. Torbensen, K., Ristori, S., Rossi, F., Abou-Hassan, A.: Tuning the chemical communication of oscillating microdroplets by means of membrane composition. *J. Phys. Chem. C* **121**(24), 13256–13264 (2017)
24. Field, R.J., Noyes, R.M.: Oscillations in chemical systems. IV. Limit cycle behavior in a model of a real chemical reaction. *J. Chem. Phys.* **60**, 1877–1884 (1974)
25. Field, R.J., Körös, E., Noyes, R.M.: Oscillations in chemical systems. II. Thorough analysis of temporal oscillation in bromate-cerium-malonic acid system. *J. Am. Chem. Soc.* **94**, 8649–8664 (1972)
26. Tyson, J.J.: A quantitative account of oscillations, bistability, and traveling waves in the Belousov-Zhabotinskii reaction. In: Field, R.J., Burger, M. (eds.) *Oscillations and Traveling Waves in Chemical Systems*, pp. 93–144. Wiley-Interscience, New York (1985)
27. Tyson, J.: Scaling and reducing the Field-Körös-Noyes mechanism of the Belousov-Zhabotinskii reaction. *J. Phys. Chem.* **81**(86), 3006–3012 (1982)
28. Burrage, K., Cardone, A., D’Ambrosio, R., Paternoster, B.: Numerical solution of time fractional diffusion systems. *Appl. Numer. Math.* **116**, 82–94 (2017)
29. Cardone, A., D’Ambrosio, R., Paternoster, B.: Exponentially fitted IMEX methods for advection-diffusion problems. *J. Comput. Appl. Math.* **316**, 100–108 (2017)
30. Cardone, A., D’Ambrosio, R., Paternoster, B.: High order exponentially fitted methods for Volterra integral equations with periodic solution. *Appl. Numer. Math.* **114C**, 18–29 (2017)
31. D’Ambrosio, R., Moccaldi, M., Paternoster, B.: Adapted numerical methods for advection-reaction-diffusion problems generating periodic wavefronts. *Comput. Math. Appl.* **74**(5), 1029–1042 (2017)
32. D’Ambrosio, R., Paternoster, B.: Numerical solution of reaction-diffusion systems of $\lambda - \omega$ type by trigonometrically fitted methods. *J. Comput. Appl. Math.* **294**, 436–445 (2016)
33. Ixaru, L.G., Paternoster, B.: A conditionally p-stable fourth-order exponential-fitting method for $y'' = f(x, y)$. *J. Comput. Appl. Math.* **106**(1), 87–98 (1999)
34. Ixaru, L.G., Berghe, G.V.: *Exponential Fitting*. Kluwer Academic Publishers, Dordrecht (2004)
35. Voorstuijts, V., Kevrekidisc, I.G., De Deckerab, Y.: Nonlinear behavior and fluctuation-induced dynamics in the photosensitive Belousov-Zhabotinsky reaction. *Phys. Chem. Chem. Phys.* **19**, 22528–22537 (2017)

Biological Systems



Understanding Embodied Cognition by Building Models of Minimal Life Preparatory Steps and a Preliminary Autopoietic Framework

Luisa Damiano¹(✉)  and Pasquale Stano²(✉) 

¹ Epistemology of the Sciences of the Artificial Research Group (ESARG),
Department of Ancient and Modern Civilizations, University of Messina,
Messina, Italy

luisa.damiano@unime.it

² Department of Biological and Environmental Sciences and Technologies
(DiSTeBA), University of Salento, Lecce, Italy

pasquale.stano@unisalento.it

Abstract. A novel scenario is emerging from the synthetic biology advancements of the last fifteen years. We refer to a well-defined multi-disciplinary sci-tech arena dedicated to the construction of biological-like systems, and, in particular, microscopic cell-like systems. The challenge of assembling a minimal cell from separated parts is generally considered the Holy Grail of biology. However, an accurate analysis of this emerging line of research, grounded in the theory of autopoiesis and its implications, is able to show its potentially high relevance for two other fields – artificial life and artificial intelligence. In this paper we intend to propose this perspective. Based on the critical discussion of recent trends and experimental results in synthetic biology, we sketch out how current research in this field can impact not only artificial life, but also artificial intelligence inquiries, in particular with respect to embodied cognition.

1 The Vision

In 1943 Arturo Rosenblueth, Norbert Wiener and Julian Bigelow inaugurated the era of cybernetics with the article *Behavior, Purpose and Teleology* [42]. They concluded it with a visionary remark, which assigns to synthetic biology (SB) a potentially crucial role in the scientific modeling of cognition.

“If an engineer were to design a robot, roughly similar in behavior to an animal organism, he would not attempt at present to make it out of proteins and other colloids. He would probably build it out of metallic parts, some dielectrics and many vacuum tubes. The movements of the robot could readily be much faster and more powerful than those of the original organism. Learning and memory, however, would be quite rudimentary. In future years, as the knowledge of colloids and proteins increases, future engineers may attempt the design of robots not only with a behaviour, but

also with a structure similar to that of a mammal. The ultimate model of a cat is of course another cat, whether it be born of still another cat or synthesized in a laboratory.”

Today, while engineering research on mechanical robots is impressively advancing, biology is integrating fifty years of progress in chemistry, biophysics and life sciences to proceed in the direction indicated by these pioneers of cybernetics. This is a way leading to bio-robots, and maybe, one day, even to approximations of what they called “ultimate” models of living beings.

Starting from 1953, the *annus mirabilis* of biology (mainly due to the James Watson’s and Francis Crick’s discovery of DNA helical structure, the completion of the insulin sequencing by Fred Sanger, and the famous experiment of Stanley Miller and Harold Urey), our understanding of “colloids and proteins” increased at a stunning level, especially in the last decades. After having explored biological systems according to the analytic or ‘taking apart’ methodology [22], in the early 2000 a new wave of biological studies emerged, explicitly inspired to engineering. This is SB, a branch of biology aiming at *constructing*, or ‘putting together’ [22], for engineering purposes, biological parts, devices and systems that do not currently exist in the natural world. More precisely, SB targets the design and the construction of programmable synthetic cells by developing first, and using later, parts-devices-systems that ultimately exploit the computational capability of molecules [3]. SB’s main research activity is based on the availability of powerful and often high-throughput bio-analytical techniques, on the progress in synthetic capability (synthesis of genes), and on the inclination of a young generation of scientists to blend biology and engineering.

However, SB also has another facet. It combines biology and engineering in order to contribute to the scientific understanding of life. This is one of the reasons for which SB stimulates critical philosophical explorations of its epistemological and theoretical approaches [30,35,38]. The attention is primarily focused on the way in which SB generates knowledge, based not on developing abstract representations, but on creating material models – that is, concrete physico-chemical models – of the biological processes under inquiry, and of the underlying biological mechanisms. Such an “understanding-by-building” methodological approach [7,39] has a significant long tradition. It was introduced by proto-cybernetic movements between the 1910s and the 1930s for the modeling of biological and cognitive processes through mechanical artifacts (hardware models). It has become a recognized scientific method with the birth of cybernetics in the 1940s. Since the 1950s it has been developed by classical artificial intelligence (AI) and, since the late 1980s, by artificial life (AL) through computer simulations (software models). With the raise of “embodied AI” [39] it has been implemented through new generations of mechanical robots (again, hardware models).

One of the main novelties generated by SB is that now we can build chemical models (wetware models) of natural processes at the molecular, supramolecular and cellular levels, because, as Rosenblueth, Wiener and Bigelow foresaw in 1943 [42], our “knowledge of colloids and proteins” has critically increased in the past years, and SB has developed a variety of “constructive” approaches [21].

In this essay we will discuss with a certain detail one of these approaches. This is the so-called *bottom-up semi-synthetic approach*, which was introduced in the early 1990s by emergent research on the origins of life, and, in particular, by the seminal work of Pier Luigi Luisi, Peter Walde and Thomas Oberholzer at the ETH-Zürich [28]. The article is composed of three parts, respectively dedicated to: (i) theory and construction of semi-synthetic minimal cells; (ii) chemical signaling between synthetic cells and biological cells; (iii) relevance and implications of the recent advancements in SB for the creation of fecund cross-disciplinary research embracing SB and AL, and in particular SB and AI. The theoretical framework defining our perspectives on these developments relies on autopoietic cognitive biology [31–33] and (related radical trends in) embodied cognitive science [51].

2 Theory and Construction of Semi-synthetic Minimal Cells

Mainstream SB operates according to the bio-brick philosophy (parts.igem.org), which refers to the construction of molecular devices – generally genetic circuits – as the assembly of standardized parts, whereby a part is generally a DNA sequence. Just like electronic engineers design and build devices from electronic parts [14, 26], synthetic biologists operate with genes and regulatory proteins, and plug them in the biological chassis, *i.e.*, the cell with its core set of genetic-metabolic circuitry.

A new wave in SB considers instead the possibility of constructing simplified (minimal) cells [27] from scratch. These synthetic cells should mimic biological cells, and thus should be capable of performing target living functions, despite the strong reduction of complexity required for their construction. They somehow resemble primitive (ancient) cells before the complexification generated by evolution. These cell-like compartments, best called semi-synthetic minimal cells (SSMCs), are interesting in many respects, such as:

1. in general, they allow life sciences to understand biochemical and biophysical processes without the interference of the other cellular processes in background;
2. in the so-called “origins-of-life research”, they help to study and understand basic mechanisms leading to the transition from inanimate to living matter;
3. in SB, they can be used to develop systems performing specific functions;
4. they are composed using dozens of separated, yet well-characterized, molecular parts (proteins, nucleic acids, lipids, etc.);
5. they can be built in the laboratory according to a novel technology based on the convergence of cell-free systems, liposome technology, and microfluidics;
6. a precise mathematical modeling can be applied to SSMCs processes;
7. despite their simplicity, SSMCs have several cell-like properties (though in a rudimentary form) and might display emergent properties;
8. although creating *living* SSMCs is the challenging long-term goal, non-living SSMCs work well for most biotechnological applications;

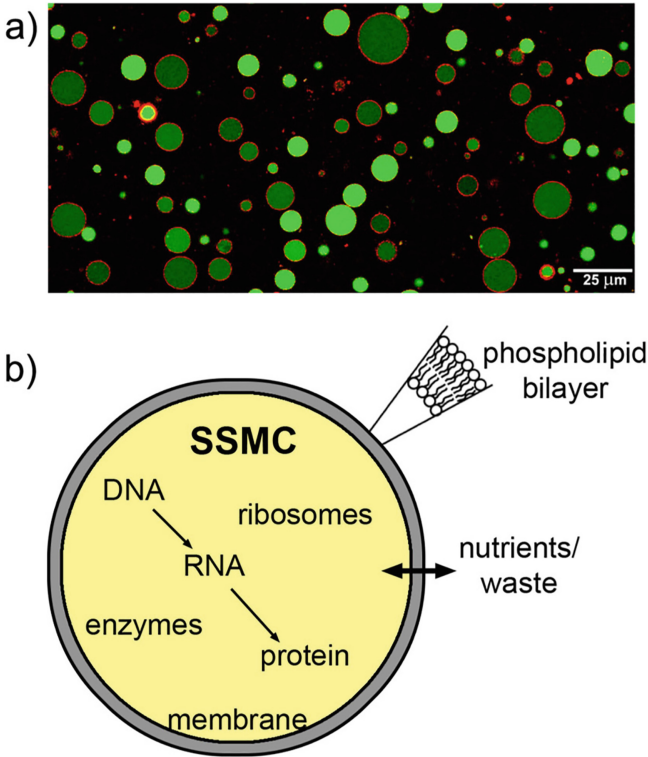


Fig. 1. SSMCs made of solute-containing liposomes. (a) Giant lipid vesicles are often used for constructing cell-like systems. The picture shows calcein-filled vesicles whose membranes have been stained by Trypan Blue. Reproduced from [48] according to the CC-BY license. (b) The minimal number of genes, enzymes, RNAs, and low molecular-weight compounds are encapsulated into synthetic lipid-based compartments, such as in the case of lipid vesicles. The membrane acts as a boundary to confine the interacting internalized molecules and allows selective passage of some molecules (nutrients, waste).

9. SSMCs might represent a concrete attempt to experimentally investigate biological autonomy and its expression in cognition [4], in particular within the autopoietic framework [31–33].

The construction of SSMCs is generally acknowledged as the SB bottom-up approach, as opposite to the mainstream approach, which is dubbed as top-down SB (for a critical discussion, see [46]). For constructing SSMCs, a minimal number of biological macromolecules (DNA, ribosomes, enzymes) are encapsulated inside liposomes (Fig. 1). Early attempts started in the 1990s. Today SSMCs can be constructed in a rather complex way and can produce proteins [47]. SSMCs easily host diverse biochemical reactions, and thus can perform simple biological-like functions.

Autopoiesis [32] offers a theoretical perspective that, as we think, best guides SSMCs design [29]. This theory deals with the question “What is life?”, and finds the answer in the specific form of dynamical organization that, according to its authors, Humberto Maturana and Francisco Varela, characterizes and defines as such all biological systems. Autopoiesis proposes that:

1. the distinctive property of living systems is their autopoiesis (namely, self-production), *i.e.*, their capability of producing and maintaining their material identity (themselves) by producing their own components (*via* metabolism);
2. since autopoiesis is a global property, its realization does not rely on components of the living systems as taken separately, or specific parts or centers within these systems, but on the way in which the components are organized within the living systems;
3. in its minimal manifestation, given at the level of minimal cells, the autopoietic organization is a self-regenerating network of operations of synthesis and destruction of components (metabolism), which: (*i*) produces its material components; (*ii*) defines by itself its topological limits through the creation of a material separation from the external environment; (*iii*) maintains itself as a unit by compensating environmental perturbations through self-regulation;
4. the self-regulative adaptive activity of autopoietic systems can be interpreted as a cognitive activity, consisting in generating internal meanings, expressed in schemes of self-regulation, for external events perceived as perturbations [10];
5. this view entails that minimal cognition is rooted in the minimal biological body, and is a radical form of embodied cognition [53].

Early work was dedicated to realize simple systems, like self-reproducing reverse micelles [2] and vesicles [52], which, owing to their simplicity and to the exploitation of surfactants self-organization, displayed dynamics that were considered similar to the autopoietic ones [31]. Moving from these simple chemical systems to more complicated cell-like ones, like the SSMCs, the attempts of creating full-fledged autopoietic systems are extremely difficult to implement. This is due to the fact that the production of SSMCs components (proteins, nucleic acids, ...) relies on complex biochemical mechanisms based on macromolecular catalysts, the enzymes, which, according to the autopoietic theory, have to be produced autonomously by the system.

To be more specific, current SSMCs are assembled by combining liposome technology and cell-free technology [5, 20, 47], for example by incorporating the PURE system (a transcription-translation system [44]) inside liposomes. The resulting cell-like system can synthesize one or more proteins and thus perform specific functions. However, what we can conceive as a genuine ‘autopoietic SSMC’ should produce, together with a target protein, also all the components of the PURE system that are involved in the synthesis of the protein. Moreover, it should be able to conservatively interact with its environment, in such a way to fuel its internal self-productive dynamic and maintain it stable through self-regulation. This behavior is currently too complex to be synthetically generated.

Although building an autopoietic synthetic cell from scratch remains the principal purpose of bottom-up SB, any ‘intermediate’ cell-like structures that can

be constructed in the lab are highly relevant. Yet not displaying an autopoietic organization, they are *per se* interesting milestones to reach, both to contribute to our understanding-by-building undertaking, and to develop novel tools for biotechnology.

Among the most interesting non-autopoietic SSMCs, we find those capable of exchanging chemical signals with biological cells, discussed below.

3 Chemical Signaling Between Synthetic Cells and Biological Cells

Synthetic cells can be built in order to recognize a chemical signal and behave accordingly, as it happens between biological cells. This is generally discussed in terms of performing a logical operation (*i.e.*, computing) [13,17]. This computational interpretation of the behavior of synthetic cells is increasingly often considered as opening a promising route to the novel bio-chemical information and communication technologies (bio-chem ICTs). However, other interpretations are possible. One of them, as we already remarked [49], is an autopoietic interpretation, in line with which these signaling mechanisms are better interpreted as attempts of autopoietic systems to conservatively (*i.e.*, ‘self-regulatively’) react to externally generated perturbations. In 2012 [49] we specified that:

“... [autopoietic systems] can perceive some external variations as perturbations of their internal process of self-production. Besides, they can react to them through an activity of self-regulation, that is, through changes in their elementary processes that compensate the alteration. In this sense, these systems can be conceived able of generating internal operational meanings for the perceived external variations. These meanings are expressed in terms of dynamical schemes of self-regulation, which externally appears as actions oriented to conservation (*e.g.*, absorbing a molecule of sugar, overcoming an obstacle...). This ‘meaning generation’ behavior – for Maturana and Varela the basic ‘cognitive’ behavior – grounds what the two researchers called “structural coupling” with the environment: a dynamic of reciprocal perturbations and compensations, in which the autopoietic system continuously generate and associate to exogenous variations operational meanings of self-regulation that allows it to keep its process of self-production in an ever-changing environment.”

Developing this autopoietic perspective, the capabilities of recognition that appear intrinsic to the molecular domain can be employed by SB to create experimental scenarios to synthetically study minimal cognition. In view of this goal, the first step is that of building ‘signaling synthetic cells’, that is, synthetic cells capable of exchanging chemical signals, or, more in general, of expressing adaptive responses to perturbations. The second step is that of using them to test, provide feedback and eventually further develop the autopoietic description of embodied cognition (Fig. 2).

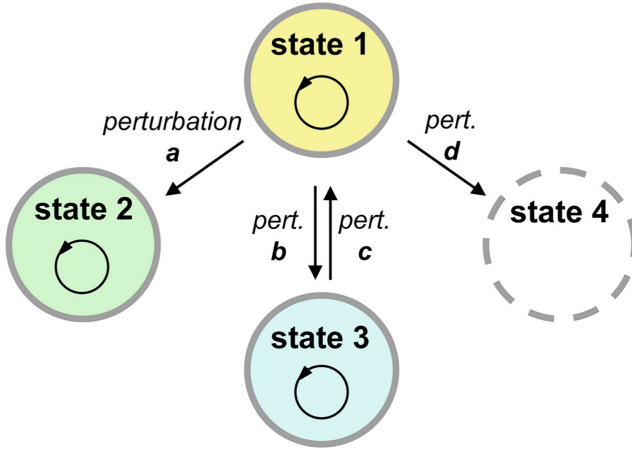


Fig. 2. Target for SSMCs and minimal cognition: generation and association of internal self-regulation patterns to external perturbative events. The transition between several states (1, 2, 3) of autopoietic cells is triggered by environmental perturbations (a, b, c) acting on the internal dynamical organization (genetic-metabolic circuitry). When the system cannot cope with a perturbation (d), it undergoes a transition toward a non-autopoietic state (4). To experimentally build this kind of artificial model of a minimal autopoietic unit equates to have the possibility of developing the autopoietic perspective on cognitive coupling by trying to answer research questions such as the following. At what level of internal dynamic complexity a minimal autopoietic unit is able to establish a relation of structural coupling with the environment as it is described by the theory of autopoiesis? Can this kind of relationship be established by autonomous systems that cannot be defined as full-fledged autopoietic systems? Why? Are there significant variations in this coupling, and the related activity of generation of meanings, with the progressive increase of the dynamic complexity of the modeled autopoietic unit? If yes, can we distinguish and classify different kinds of cognitive coupling already at the level of minimal synthetic autopoietic units? Can we do that with regard to the coupling that can be established between different artificial, or between artificial and natural, minimal autopoietic units?

An analysis of the deep implications of this approach exceeds the scope of this contribution, and we will develop it in future publications. Here we would like to underline that, as remarked also in Sect. 4, these developments might profoundly impact the epistemological and theoretical frameworks of AI research.

Coming back to traditional bio-chem ITCs, the relevance of inter- and intracellular molecular signaling has been put forward by Tadashi Nakano [37], and application to nanomedicine has been lucidly defined by Philip LeDuc [23] (Fig. 3). Genetic, regulatory and metabolic circuitry could be adapted to this goal (consider for instance SSMCs endowed with plugged-in circuits).

Related experimental work has been carried out in recent years. In 2009 Ben Davis and collaborators reported that chemicals inside vesicles could synthesize a sugar-like molecule that, when released in the medium and converted to a

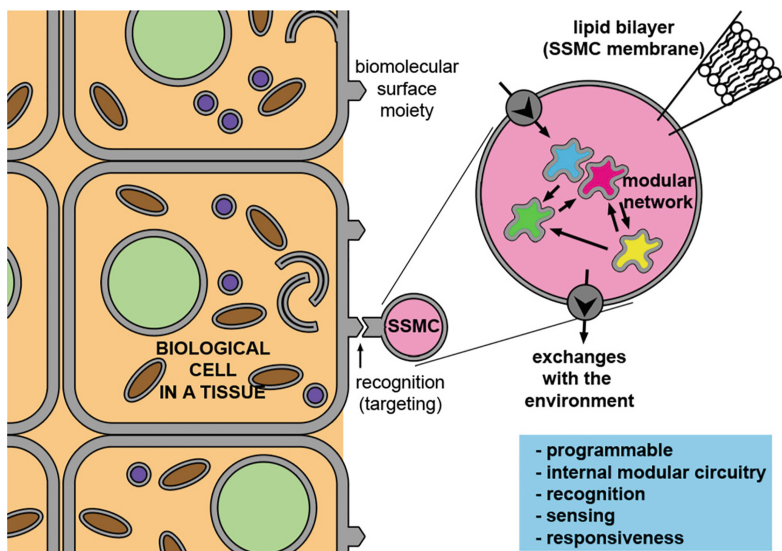


Fig. 3. By advancing the current biotechnology for SSMCs construction, potential medical applications of more sophisticated and programmable synthetic cells can be envisaged. A goal would be to construct cell-like systems that, once injected in the human body, reach a specific target region and, thanks to chemical information processing, are able to act appropriately, for example producing *in situ* a cytotoxic drug or a stimulus to trigger a cellular response. This has been referred as the concept of “pseudo-cell factories” or “nanofactories” (illustrated above) [23].

borate derivative, can stimulate a response in receiving cells (*Vibrio harveyi* bacteria) [16]. Our group is involved in this kind of research since 2012, and has been promoting in this area a SB approach [34, 41, 49]. Recently we reported an effective chemical signaling between SSMCs and *Pseudomonas aeruginosa* [40]. In 2014 Sheref Mansy built synthetic cells acting as “translators” for *Escherichia coli* [25], and more recently reported a two-way chemical exchange of signals between synthetic cells and bacteria [24]. Chemical signaling between synthetic cells has been also reported by Kate Adamala [1] and by Sheref Mansy with Stephen Mann [50].

A common point of this scientific literature is that it is actively paving the way to a scientific exploration of natural processes of communicational exchange. Based on the understanding-by-building methodology, this scientific undertaking starts from the synthetic exploration of minimal and mono-directional molecular signaling processes, and progressively targets higher levels of complexity. These include bi-directional and multi-directional exchanges, as well as different forms of behavioral coordination between cells, in order to define the conditions of emergence of full-fledged communication.

3.1 A Turing Test for Synthetic Cellularity?

As we discussed in previous works [6, 41], an interesting and somehow provocative scenario was proposed in 2006 by Lee Cronin, Natalio Krasnogor, Ben Davis and collaborators [8]. These authors argued that a sort of Turing test, targeting minimal communication skills, in the sense specified above, could help to determine whether a cell-like system can be considered alive. The advantage provided by this test would be that of bypassing the issue of defining and clarifying what life is, in analogy with the approach of the classical Turing Test – determining whether a system is intelligent, while bypassing the problem of defining and clarifying what intelligence is.

As related literature pointed out, in AI the artificial re-creation of a target cognitive behavior can be – and often is – mere imitation: the recreation of the phenomenology of the behavior based on biologically implausible mechanisms, which would abrogate *de facto* the actual relevance of such a test [9].

With regard to this, the case of SSMCs is different. Due to their constitutive molecular nature, they are able to increment the possibilities of generating plausible mechanisms, those found in nature, and the “imitation game” might be significant. The experimental scenarios mentioned above, based on the understanding-by-building approach, could prove useful to address interesting technical, theoretical, epistemological, and philosophical questions. At this minimal level, will the synthetic/artificial cells reproduce the ‘cognitive pattern’ of a natural/biological partner? Will SB blur, or even break, the synthetic *vs.* natural divide? [6].

It is interesting to shortly consider the attempt done by Sheref Mansy and colleagues [24] of *quantifying* the life-likeness (or better, *Vibrio fischeri*-likeness) of synthetic cells capable of sending and receiving signals to/from the bacterium *V. fischeri*. The quantification was done on the basis of RNA sequencing (*i.e.*, determining the gene expression profile of natural cells in response to the activity of the synthetic cells). From a series of comparative tests (*V. fischeri vs. V. fischeri*, *V. fischeri vs. non-functional synthetic cells*, and *V. fischeri vs. functional synthetic cells*), surprisingly their synthetic cells scored 39% life-like. However, as the authors remarked, the genetically encoded elements in signaling synthetic cells were only two (LuxR and LuxI), whereas the calculus was extended to include the more than 100 genes encoding for the transcription-translation machinery. Thus the result (39%) refers to an ideal synthetic cell that autonomously produces all its proteins and nucleic acid. As mentioned above, such a system is still out of reach.

4 SB-AI

The cross-disciplinary connection between SB and AI, based on the understanding-by-building approach, is a task that frontier research is starting to address [11]. Can SB, and in particular research on synthetic cells, be useful in AI inquiries? In principle, the answer is positive in relation to the new embodied AI. This, differently from classic AI, acknowledges a deep integration – in its radical

versions, a unity – between cognitive and biological processes. We refer in particular to radical approaches to embodied cognition, which, in line with the autopoietic theory, ground the cognitive mind in the biological processes of construction and maintenance of the biological body. Within these explorative contexts the synergy of SB and AI through the understanding-by-building approach appears as a solid possibility. Research on minimal cells, by constructing material (chemical) models of minimal living systems, opens the concrete prospect of experimental scenarios to scientifically explore minimal embodied cognitive and related processes (organizational closure, self-distinction, self-production, self-maintenance, self-regulation, dynamical coupling between autonomous or autopoietic systems and their environments, molecular signaling between autonomous units, behavioral coupling between autonomous units, among others). All these processes stem from the molecular dimension, molecular dynamics, strength of entropy at the molecular level, molecular energies and interplay with the thermal background [19], which cannot be found in macroscopic objects. It is not by chance that life originated at the molecular and supra-molecular level. Following this line of reasoning, it is worth noticing that molecular systems significantly differ from electronic computers. Their operations can be interpreted as forms of computing, but with reference to properties that diverge from the one we are used to consider related to computation. These different properties are well illustrated in Nakano’s work [36, 37].

Since a few years, we have been engaged in promoting the possibility of a synergic cross-fertilization between SB and AI [11, 45] focused on minimal embodied cognition. In what follows, we draw the main lines of a possible SB-AI approach to the study of minimal cognition based on the theory of autopoiesis – an approach that we call ‘Chemical Autopoietic AI’.

4.1 Chemical Autopoietic AI: Drawing the Basic Lines of a SB-AI Research Program

The potentialities that autopoiesis can express in AI rely on its definition of life, based on two main reasons. The first refers to the “synthetic” nature of this definition. Maturana and Varela’s answer to the question “What is life?” is not analytical, as traditional definitions of life are. Autopoiesis answers this question not by proposing a list of properties of living systems, but by theoretically defining a mechanism able to generate, from a multiplicity of separated components, a minimal living system potentially able to produce the whole biological domain as we know it. The idea of providing a “synthetic” definition of life is essentially this: theoretically determining a mechanism able to generate, from scratch, the whole biological phenomenology. Its interest for the sciences of the artificial, and AL and AI in particular, is evident. It promises that, if science implements the mechanism specified by the autopoietic definition of life, then, in principle, it will be able to re-create all biological processes. And these processes, according to Maturana and Varela, are intrinsically cognitive processes. In the words of the pioneers of cybernetics: providing a material model of the autopoietic network would open to the sciences of the artificial the possibility

of creating “ultimate” models of living and cognitive systems – *i.e.*, artificial, but genuinely living and cognitive systems. The second reason of the relevance of the autopoietic definition of life for AI relies on its theoretical content. This corresponds to the notion of autopoietic organization, which describes, in the following terms, a mechanism able to generate minimal and cognitive systems - and, through them, all biological and cognitive systems.

[...] A network of process of production (transformation and destruction) of components that produces the components which: (i) through their interactions and transformations continuously regenerate and realize the network of processes (relations) that produced them; and (ii) constitute it (the machine) as a concrete unity in the space in which they (the components) exist by specifying the topological domain of its realization as such a network [...] [32] (p. 79).

One of the peculiarities of the notion of autopoietic organization, as it clearly emerges from its definition, is that it is independent from the designation of specific components. This implies that, in principle, the sciences of the artificial engaged in the project of creating material models of the autopoietic organization do not have to focus on the actual components of life as we know it. They can use all kinds of components that prove able to generate the network described by the autopoietic definition of life. This gives AL and AI the possibility to implement different material models of (minimal) life and cognition, that is, (a variety of) forms of biological and cognitive processes that, with respect to their material structure, do not exist in nature. These two characteristics of the autopoietic definition of life – its synthetic character and its independence from specific components – make it particularly interesting for SB research on synthetic cells. In particular, they make it appealing for the subsections of SB focusing on the convergence between liposome technology and cell-free systems, with respect to which the theory of autopoiesis defines the long-term goal of building minimal living systems in the laboratory.

This goal allows SB to aspire to actively contribute to AI research, with a significant advantage with regard to other ways of modeling life and cognition based on autopoiesis. Computer simulations can provide only abstract artificial models of the autopoietic definition of life, and current mechanical robotics appears to be far from the possibility of generating material models of the dynamic network that this definition describes [15]. Differently from them, SB operates with components that are chemical molecules, and thus can differ from those of terrestrial life for chemical structure, but not for reactivity in general terms. Thus, in principle, SB is able to generate embodied models, *i.e.*, chemical models, of this kind of network, and actually is already engaged in designing primitive versions of them. Its main obstacle is the very high level of complexity of even minimal autopoietic networks – especially when based on available biomacromolecules. As we will show in a future publication, however, this does not preclude the possibility of building simplified versions of the autopoietic organization, based on theoretical re-elaborations of the original autopoietic definition of life.

The promises of this kind of SB-AI approach would be extremely relevant for the evolution of AI, for a series of reasons:

1. As we will show elsewhere, this approach would provide AI with experimentally explorable material models of an “intrinsically intentional” cognitive agent. We refer to a cognitive agent whose relations with the environment are charged with “intrinsic” meanings, depending on the conservation of its organization and its ways of existence. As relevant literature emphasizes since Hubert Dreyfus’ “What Computers Cannot Do” (1979) [12], John Searle’s “Chinese Room Argument” (1980) [43], and Stevan Harnad’s “Symbol Grounding Problem” (1990) [18], this is one of the main weak point of (both traditional and embodied) AI [53].
2. The experimental exploration of this notion through its basic wetware modelization could include not only the structural coupling of the minimal synthetic autopoietic system with its environment, but also the dynamics of interactions with this niche that could lead it to develop higher levels of organizational complexity, as implied by the “synthetic definitional approach” characterizing the autopoietic theory.
3. Both explorations could have applicative implications leading to hybrid bio-mechanical robots and, more in general, synthetic cognitive systems based on the autopoietic approach, and/or, more in general, a radical approach to embodied cognition.

The above described difficulties in fully implementing this SB-AI research line are currently engaging us in developing a simplified version of it. We plan to fully describe it in future works as a first step in the inauguration of the Chemical Autopoietic AI approach.

Acknowledgments. The authors thank Pier Luigi Luisi (Roma Tre University and ETH Zürich) for inspiring discussions. This work has been stimulated by our involvement in the European COST Action CM-1304 “*Emergence and Evolution of Complex Chemical Systems*” and TD-1308 “*Origins and evolution of life on Earth and in the Universe (ORIGINS)*”.

References

1. Adamala, K.P., Martin-Alarcon, D.A., Guthrie-Honea, K.R., Boyden, E.S.: Engineering genetic circuit interactions within and between synthetic minimal cells. *Nat. Chem.* **9**(5), 431–439 (2017)
2. Bachmann, P.A., Walde, P., Luisi, P.L., Lang, J.: Self-replicating reverse micelles and chemical autopoiesis. *J. Am. Chem. Soc.* **112**(22), 8200–8201 (1990)
3. Benenson, Y., Gil, B., Ben-Dor, U., Adar, R., Shapiro, E.: An autonomous molecular computer for logical control of gene expression. *Nature* **429**(6990), 423–429 (2004)
4. Bich, L., Damiano, L.: Life, autonomy and cognition: an organizational approach to the definition of the universal properties of life. *Orig. Life Evol. Biosph.* **42**, 389–397 (2012)

5. Blain, J.C., Szostak, J.W.: Progress toward synthetic cells. *Annu. Rev. Biochem.* **83**(1), 615–640 (2014)
6. Bracciali, A., Cataldo, E., Damiano, L., Felicioli, C., Marangoni, R., Stano, P.: From cells as computation to cells as apps. In: Gadducci, F., Tavosanis, M. (eds.) *HaPoC 2015. IAICT*, vol. 487, pp. 116–130. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47286-7_8
7. Cordeschi, R.: *The Discovery of the Artificial*. Springer, Heidelberg (2002). <https://doi.org/10.1007/978-94-015-9870-5>
8. Cronin, L., Krasnogor, N., Davis, B.G., Alexander, C., Robertson, N., Steinke, J.H.G., Schroeder, S.L.M., Khlobystov, A.N., Cooper, G., Gardner, P.M., Siepmann, P., Whitaker, B.J., Marsh, D.: The imitation game—a computational chemical approach to recognizing life. *Nat. Biotechnol.* **24**(10), 1203–1206 (2006)
9. Damiano, L., Hiolle, A., Cañamero, L.: Grounding synthetic knowledge. In: Lenaerts, T., Giacobini, M., Bersini, H., Bourguine, P., Dorigo, M., Doursat, R. (eds.) *Advances in Artificial Life, ECAL 2011*, pp. 200–207. MIT Press, Cambridge (2011)
10. Damiano, L.: Co-emergences in life and science: a double proposal for biological emergentism. *Synthese* **185**(2), 273–294 (2012)
11. Damiano, L., Kuruma, Y., Stano, P.: What can synthetic biology offer to artificial intelligence (and vice versa)? *Biosystems* **148**, 1–3 (2016)
12. Dreyfus, H.L.: *What Computers Can't Do: The Limits of Artificial Intelligence*. Harper and Row, New York (1979). Revised edition
13. Elowitz, M.B., Leibler, S.: A synthetic oscillatory network of transcriptional regulators. *Nature* **403**(6767), 335–338 (2000)
14. Endy, D.: Foundations for engineering biology. *Nature* **438**, 449–453 (2005)
15. Froese, T., Ziemke, T.: Enactive artificial intelligence: investigating the systemic organization of life and mind. *Artif. Intell.* **173**(3), 466–500 (2009)
16. Gardner, P.M., Winzer, K., Davis, B.G.: Sugar synthesis in a protocellular model leads to a cell signalling response in bacteria. *Nat. Chem.* **1**(5), 377–383 (2009)
17. Gardner, T.S., Cantor, C.R., Collins, J.J.: Construction of a genetic toggle switch in *Escherichia coli*. *Nature* **403**(6767), 339–341 (2000)
18. Harnad, S.: The symbol grounding problem. *Physica D* **42**, 335–346 (1990)
19. Hoffmann, P.M.: *Life's Ratchet: How Molecular Machines Extract Order from Chaos*, 1st edn. Basic Books. A Member of the Perseus Books Group, New York (2012)
20. Ichihashi, N., Matsuura, T., Kita, H., Sunami, T., Suzuki, H., Yomo, T.: Constructing partial models of cells. *Cold Spring Harb. Perspect. Biol.* **2**(6), a004945 (2010)
21. Kaneko, K.: *Life: An Introduction to Complex Systems Biology. Understanding Complex Systems*. Springer, Heidelberg (2006). <https://doi.org/10.1007/978-3-540-32667-0>
22. Langton, C.G.: Artificial life. In: Boden, M.A. (ed.) *The Philosophy of Artificial Life*, pp. 39–94. Oxford University Press, Oxford (1996)
23. LeDuc, P.R., Wong, M.S., Ferreira, P.M., Groff, R.E., Haslinger, K., Koonce, M.P., Lee, W.Y., Love, J.C., McCammon, J.A., Monteiro-Riviere, N.A., Rotello, V.M., Rubloff, G.W., Westervelt, R., Yoda, M.: Towards an in vivo biologically inspired nanofactory. *Nat. Nanotechnol.* **2**, 3–7 (2007)
24. Lentini, R., Martín, N.Y., Forlin, M., Belmonte, L., Fontana, J., Cornella, M., Martini, L., Tamburini, S., Bentley, W.E., Jousson, O., Mansy, S.S.: Two-way chemical communication between artificial and natural cells. *ACS Cent. Sci.* **3**(2), 117–123 (2017)

25. Lentini, R., Santero, S.P., Chizzolini, F., Cecchi, D., Fontana, J., Marchioretto, M., Del Bianco, C., Terrell, J.L., Spencer, A.C., Martini, L., Forlin, M., Assfalg, M., DallaSerra, M., Bentley, W.E., Mansy, S.S.: Integrating artificial with natural cells to translate chemical messages that direct *E. coli* behaviour. *Nat. Commun.* **5**, 4012 (2014)
26. de Lorenzo, V., Danchin, A.: Synthetic biology: discovering new worlds and new words. *EMBO Rep.* **9**(9), 822–827 (2008)
27. Luisi, P.L., Ferri, F., Stano, P.: Approaches to semi-synthetic minimal cells: a review. *Naturwissenschaften* **93**, 1–13 (2006)
28. Luisi, P.L., Walde, P., Oberholzer, T.: Lipid vesicles as possible intermediates in the origin of life. *Curr. Opin. Colloid Interface Sci.* **4**(1), 33–39 (1999)
29. Luisi, P.L.: Autopoiesis: a review and a reappraisal. *Naturwissenschaften* **90**(2), 49–59 (2003)
30. Luisi, P.L.: The synthetic approach in biology: epistemological notes for synthetic biology. In: Luisi, P.L., Chiarabelli, C. (eds.) *Chemical Synthetic Biology*, pp. 343–362. Wiley, Chichester (2011)
31. Luisi, P., Varela, F.: Self-replicating micelles - a chemical version of a minimal autopoietic system. *Orig. Life Evol. Biosph.* **19**, 633–643 (1989)
32. Maturana, H.R., Varela, F.J.: *Autopoiesis and Cognition: The Realization of the Living*, 1st edn. D. Reidel Publishing Company, Dordrecht (1980)
33. Maturana, H.R., Varela, F.J.: *De máquinas y seres vivos: Una teoría de la organización Biológica*. Editorial Universitaria, Santiago (1972)
34. Mavelli, F., Rampioni, G., Damiano, L., Messina, M., Leoni, L., Stano, P.: Molecular communication technology: general considerations on the use of synthetic cells and some hints from in silico modelling. In: Pizzuti, C., Spezzano, G. (eds.) *WIVACE 2014. CCIS*, vol. 445, pp. 169–189. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12745-3_14
35. Morange, M.: A critical perspective on synthetic biology. *Hyle* **15**(1), 21–30 (2009)
36. Nakano, T., Eckford, A.W., Haraguchi, T.: *Molecular Communications*. Cambridge University Press, Cambridge (2013)
37. Nakano, T., Moore, M., Enomoto, A., Suda, T.: Molecular communication technology as a biological ICT. In: Sawai, H. (ed.) *Biological Functions for Information and Communication Technologies. Studies in Computational Intelligence*, vol. 320, pp. 49–86. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-15102-6_2
38. O'Malley, M.A.: Making knowledge in synthetic biology: design meets kludge. *Biol. Theory* **4**(4), 378–389 (2009)
39. Pfeifer, R., Scheier, C.: *Understanding Intelligence*. MIT Press, Cambridge (1999)
40. Rampioni, G., D'Angelo, F., Messina, M., Zennaro, A., Kuruma, Y., Tofani, D., Leoni, L., Stano, P.: Synthetic cells produce a quorum sensing chemical signal perceived by *Pseudomonas aeruginosa*. *Chem. Commun.* **54**(17), 2090–2093 (2018)
41. Rampioni, G., Mavelli, F., Damiano, L., D'Angelo, F., Messina, M., Leoni, L., Stano, P.: A synthetic biology approach to bio-chem-ICT: first moves towards chemical communication between synthetic and natural cells. *Nat. Comput.* **13**, 1–17 (2014)
42. Rosenblueth, A., Wiener, N., Bigelow, J.: Behavior, purpose and teleology. *Philos. Sci.* **10**, 18–24 (1943)
43. Searle, J.: Minds, brains, and programmes. *Behav. Brain Sci.* **3**, 417–424 (1980)
44. Shimizu, Y., Inoue, A., Tomari, Y., Suzuki, T., Yokogawa, T., Nishikawa, K., Ueda, T.: Cell-free translation reconstituted with purified components. *Nat. Biotechnol.* **19**(8), 751–755 (2001)

45. Stano, P., Kuruma, Y., Damiano, L.: Synthetic biology and (embodied) artificial intelligence: opportunities and challenges. *Adapt. Behavior* **26**(1), 41–44 (2018)
46. Stano, P.: The birth of liposome-based synthetic biology: a brief account. In: Pearson, B.R. (ed.) *Liposomes: Historical, Clinical and Molecular Perspectives*, pp. 37–52. Nova Science Publishers, Inc. (2017)
47. Stano, P., Carrara, P., Kuruma, Y., de Souza, T.P., Luisi, P.L.: Compartmentalized reactions as a case of soft-matter biotechnology: synthesis of proteins and nucleic acids inside lipid vesicles. *J. Mater. Chem.* **21**, 18887–18902 (2011)
48. Stano, P., Mavelli, F.: Protocells models in origin of life and synthetic biology. *Life* **5**(4), 1700–1702 (2015)
49. Stano, P., Rampioni, G., Carrara, P., Damiano, L., Leoni, L., Luisi, P.L.: Semi-synthetic minimal cells as a tool for biochemical ICT. *Bio Syst.* **109**(1), 24–34 (2012)
50. Tang, T.Y.D., Cecchi, D., Fracasso, G., Accardi, D., Coutable-Pennarun, A., Mansy, S.S., Perriman, A.W., Anderson, J.L.R., Mann, S.: Gene-mediated chemical communication in synthetic protocell communities. *ACS Synth. Biol.* **7**(2), 339–346 (2018)
51. Varela, F.J., Thompson, E.T., Rosch, E.: *The Embodied Mind: Cognitive Science and Human Experience*. The MIT Press, Cambridge (1992). New edition
52. Walde, P., Wick, R., Fresta, M., Mangone, A., Luisi, P.: Autopoietic self-reproduction of fatty-acid vesicles. *J. Am. Chem. Soc.* **116**, 11649–11654 (1994)
53. Ziemke, T.: The body of knowledge: on the role of the living body in grounding embodied cognition. *Biosystems* **148**(Suppl. C), 4–11 (2016)



Computing Hierarchical Transition Graphs of Asynchronous Genetic Regulatory Networks

Marco Pedicini¹ , Maria Concetta Palumbo² , and Filippo Castiglione² 

¹ Department of Mathematics and Physics, Roma Tre University, Rome, Italy
marco.pedicini@uniroma3.it

² CNR - Institute for Applied Computing “M. Picone”, Rome, Italy

Abstract. In the field of theoretical biology the study of the dynamics of the so-called gene regulatory networks is useful to follow the relationship between the expression of a gene and its dynamic regulatory effect on the cell fate. To date, most of the models developed for this purpose, applies the synchronous update schedule while reality is far from being so. On the other hand, the more realistic asynchronous update requires to compute all possible updates at each single instant, thus bearing a much greater computational load.

In the present work, we describe a novel method that addresses the problem of efficiently exploring the dynamics of a gene regulatory network with the asynchronous update.

Keywords: SAT solver · Discrete dynamical systems
Tarjan’s algorithm · Gene regulatory networks
Strongly connected components

1 Introduction

A *gene regulatory network* (GRN) can be regarded as a discrete dynamical system with a transition function $T : S \rightarrow S$ which is determined by the activation/inhibition dependencies between a given number of genes, transcription factors or RNA molecules, where S is the set of all activation profiles of the n involved elements, that hereinafter we refer to as *the genes*. In its simplest form with the number of activation levels $q = 2$ representing genes that are either *activated* or *silent*, this function T is represented as a vector of Boolean expressions. In this and other more complicated cases (*i.e.*, $q > 2$) components of T come in special forms as polynomials and the system can be considered a Polynomial Dynamical System on Finite Fields \mathbb{F}_q [VCL12].

Given the transition function T , a dynamical system can be described as the graph consisting in the set of the ordered pairs $(s, T(s))$ for any $s \in S$. This graph is called the *state transition graph* (STG) of the dynamics T . The transition function T can be applied according to a synchronous schedule, meaning that

all genes are updated at the same time. A more general situation occurs when we want to study the evolution in the case of asynchronous dynamics: in this case, the gene to be updated is chosen at random among all genes. In both cases, the size N of the set S of the vertices of the state transition graph of a dynamics T is exponential in the number of genes n ($N = q^n$) and any exhaustive method for the investigation of its structure rapidly becomes intractable even for small n . In the asynchronous case things are more difficult in the sense that the number of arcs in the state transition graph is much larger.

In what follows we refer to the *state transition graph structure* as the set of attractors and their properties. An *attractor* is a set A of states which coincides with the set $\bigcup_k T^k(A)$ of all its successors. In particular, in the synchronous dynamics only two forms of attractors are possible: *limit cycles* (also referred to as *cyclic attractors*) and *steady states* (*point attractors*).

Whereas any steady state of the synchronous dynamics is also a steady state in the asynchronous case, the same cannot be affirmed for synchronous dynamics limit cycles. In fact, a cycle could be or not a limit cyclic in the asynchronous dynamics. Intuitively, the reason for this behaviour is that the update function is non-deterministic and therefore it allows the dynamics to exit from cycles.

A *strongly connected component* (SCC) of a state transition graph is a set of states S , such that for any pair of states $s, s' \in S$, a directed path from s to s' exists. Among all SCCs of a state transition graph we are interested in the *maximal* ones with respect to the classical subset relationship. Let's call $\mathcal{M}(T)$ the set of all maximal SCCs in the state transition graph of the dynamics T . A strongly connected component is called *terminal* if it has no outgoing edges. Note that given a state transition graph there exists at least one terminal SCC. Given two maximal components $A, B \in \mathcal{M}(T)$ we say that A *precedes* B (indicated $A \rightarrow B$) whenever a directed path from A to B exists.

Terminal SCCs are *minimal* with respect to this ordering relationship because they are the attractors of the dynamics and by definition they are their own successors. In the literature of asynchronous networks, in order to emphasize that a limit cycle cannot coincide with an attractor but rather with a part of it, attractors which are not point-attractors are sometimes referred to as *loose attractors* [HB97].

An informative representation of the state transition graph structure is then provided by a graph in which the set of nodes/vertices is the set of maximal SCCs $\mathcal{M}(T)$ and edges/arcs are the pairs (A, B) such that A precedes B ($A \rightarrow B$). In a similar way to [BCM+13], we call *hierarchical transition graph* (HTG) of the asynchronous network this compact version of the asynchronous state transition graph structure.

In this work, we are concerned with the method to determine the hierarchical transition graph under certain hypothesis. Moreover, the algorithm is based on the possibility of generating transition paths in the state transition graph that avoid to cross previously determined cycles. Despite the number of nodes of the state transition graph is exponential in the number of genes, the algorithm we propose here is efficient if the graph satisfies few necessary hypotheses (described below) in order to visit only a small fraction of the whole graph.

The best algorithm for finding SCC in a directed graph is the algorithm from Tarjan [Tar72]. Its time complexity is $o(N + M)$, *i.e.*, linear in the number of nodes N of the resulting state transition graph, and in the number of transitions between states M [Tar72]. We achieve a similar bound in the size of the hierarchical transition graph: the linear bound is therefore obtained with respect to (i) the number of maximal SCCs, (ii) the number of nodes in any SCC and (iii) the length of transient paths connecting pairs of SCCs. This result can be attained because thanks to the use of logical Boolean expression of the dynamics we do not need to explicitly compute the whole state transition graph. Instead, by following the approach by Dubrova and Teslenko [DT11], we generate transition paths as solutions of the Boolean satisfiability problem, which can be computed by using highly specialised and optimised software called *SAT solvers*. In particular, logical expressions can be adapted during the computation, in such a way that their paths avoid nodes belonging to previously-discovered cycles.

In the following, we use the term *SAT-complexity* meaning that we count any call to the SAT solver at unitary cost; this is the same situation assumed in the case of synchronous dynamics in [DT11], where the main-loop iterates on the number of cycles of the relative state transition graph. Nevertheless, from the complexity viewpoint, any call to the SAT solver could impact on runtime with an exponential cost. Therefore, strictly speaking, the procedure cannot be said to have polynomial complexity but we can measure the complexity of the procedure in terms of calls to the SAT solver and also in terms of the fraction of the state transition graph visited in order to determine the hierarchical transition graph of the asynchronous network. If the above mentioned three conditions on the structure of SCCs are respected, the proposed variant of Tarjan’s algorithm has a polynomial SAT-complexity bound on the number of genes n .

The main result of our work is the presentation of a new algorithm to determine the hierarchical transition graph of Boolean networks with several dozens of genes in the asynchronous dynamics. To this aim, we recall in Sect. 2 some definitions and formalisms on Boolean regulatory networks. In Sect. 3, we summarise the algorithm for finding limit cycles in the case of synchronous Boolean networks. Our algorithm is obtained by merging the (optimal) Tarjan approach to the determination of SCCs and the algorithm that generates paths by the SAT solver. The combination of these two approaches, described in Sect. 4, is the main result presented in this paper since it allows to determine the hierarchical transition graph of SCCs without exploring the entire state transition graph.

2 Boolean Networks as Dynamical Systems

Since Kauffman’s studies, steady states and limit cycles in gene regulatory networks are regarded as set up of cellular genetic programs. Therefore there is some interest in studying the dynamics of groups of genes in the context of biological functions they are supposed to be involved in. For instance, cell differentiation is one of these functions in which the activation of the genetic transcription program brings the cell into a novel phenotypic state [Kau93, DJ02].

Definition 1. A gene regulatory network of a set of genes V ($|V| = n$) assuming values in a scalar domain \mathbb{K} is described by components (i.e., by dependency functions)

$$f_v : \mathbb{K}^n \rightarrow \mathbb{K} \text{ for any gene } v \in V.$$

Each of these functions is one component of the synchronous global updating function of the state (that for the sake of simplicity we call the dynamics),

$$T_{\text{sync}} : \mathbb{K}^n \rightarrow \mathbb{K}^n$$

where for any state $s = (k_1, \dots, k_n) \in \mathbb{K}^n$ we obtain a new state of the system by

$$T_{\text{sync}}(s) = (f_{v_1}(s), \dots, f_{v_n}(s)).$$

In the case of Definition 1, the dynamics is termed *synchronous* to put in evidence the fact that at each step all the components of the state are updated at once. A more flexible situation occurs when we want to study the evolution given the *asynchronous* dynamics. In this case, each component is updated independently from the others with no specific order in the sequence of updates. The resulting behaviour can be thought as a non-deterministic dynamical system where starting from a configuration it is possible to reach different configurations as a consequence of the choice of the component v_i (with i chosen at random) to update.

Definition 2. The asynchronous updating function is

$$T_{\text{async}} : \mathbb{K}^n \rightarrow \mathbb{K}^n$$

where for any state $s = (k_1, \dots, k_n) \in \mathbb{K}^n$ we obtain a new state of the system by

$$T_{\text{async}}(s) = (k_1, \dots, f_{v_i}(s), \dots, k_n) \quad \text{for any choice of } i \in \{1, \dots, n\}.$$

Definition 3. The state transition graph of the dynamics T (either T_{sync} or T_{async}) is a graph $G(T)$ with nodes in \mathbb{K}^n and edges $(s, s') \in \mathbb{K}^n \times \mathbb{K}^n$ if and only if $s' = T(s)$.

Note that the number of nodes N in the state transition graph corresponds to the number of possible states which is the exponential in the size $|\mathbb{K}|$ to the power n , the number of genes, namely

$$N = |G(T)| = |\mathbb{K}|^n$$

regardless of the chosen dynamics (synchronous or asynchronous). The most commonly used scalar field consists of the binary one $\mathbb{K} = \mathbb{F}_2$, in which case each component of the dynamics T can be expressed as a Boolean expression, and the corresponding genetic regulatory network is simply referred to as a Boolean network (BN).

Example 1. Let us consider a genetic regulatory network on \mathbb{F}_2 , with $|V| = 3$ and the components given by the following Boolean expressions:

$$\begin{aligned} f_1(s_1, s_2, s_3) &= \neg s_3 \wedge (s_1 \vee s_2) \\ f_2(s_1, s_2, s_3) &= s_1 \wedge s_3 \\ f_3(s_1, s_2, s_3) &= \neg s_3 \vee (s_1 \wedge s_2) \end{aligned} \quad (1)$$

For what concerns the corresponding state transition graph, we have that $N = q^n = 2^3 = 8$. Figure 1 shows the two graphs corresponding to state transition graph in both synchronous and asynchronous dynamics.

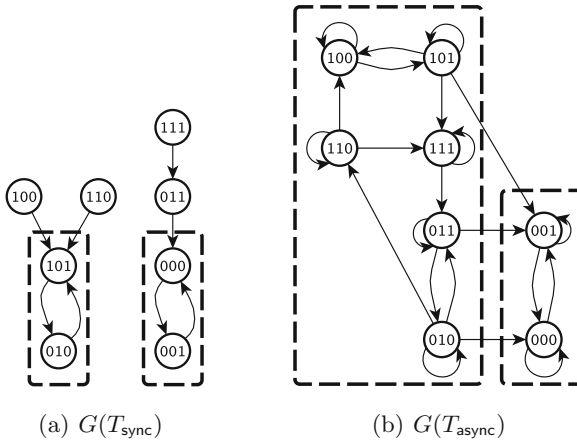


Fig. 1. Synchronous and asynchronous dynamics state transition graphs of the Boolean network specified in Eq. (1).

We are interested in several parameters which describe the state transition graph obtained starting from a given Boolean network with different transition functions and, particularly in biological applications we are concerned with the determination of the strongly connected components of the state transition graph since the biological interpretation of SCCs can be related to the stable functional characterisation of the cell behaviour [Kau93, DJ02]. Synchronous updates are rough but reasonable models of (early) response in signal networks. In Fig. 1, we compare the two state transition graphs $G(T_{\text{sync}})$ and $G(T_{\text{async}})$. In the first case, we see that any path terminates in a cycle, whilst in the latter case we have many self loops and two SCCs, one of which is terminal (*i.e.*, an attractor). When the state transition graph is available and feasible to manage, one can describe the relationship between pair of maximal SCCs as a partial order.

A useful definition in [GCBP+13] describes a “state-transition diagram” as a *hierarchical transition graph*. Hierarchical transition graphs are built on the analysis of the paths from initial states to attractors. In this paper, we tackle the

problem of determining these SCCs in the $G(T_{\text{async}})$ when the graph is too large to be explicitly computed, although its compressed form (*i.e.*, the hierarchical transition graph) can be effectively computed. See an example in Fig. 2.

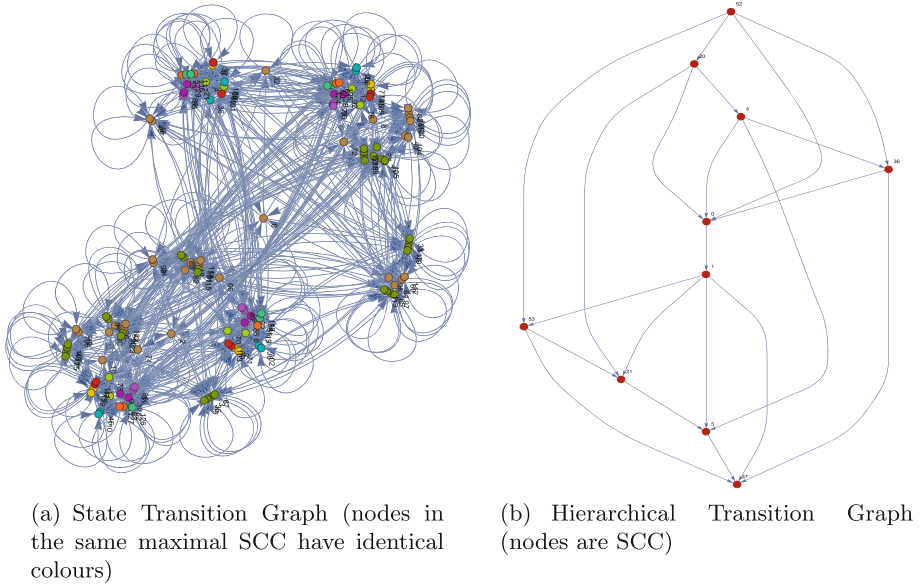


Fig. 2. An asynchronous random network with $n = 7$. The hierarchical transition graph in panel (b) (nodes are the maximal SCCs) is much more compact than the corresponding state transition graph in panel (a). (Color figure online)

Definition 4. Given the state transition graph $G(T_{\text{async}})$, $\mathcal{M}(T)_{\text{async}}$ the set of its maximal strongly connected components and \rightarrow a partial order relation between elements of $\mathcal{M}(T)_{\text{async}}$, we define the hierarchical transition graph as the graph

$$H(T_{\text{async}}) := (\mathcal{M}(T)_{\text{async}}, \rightarrow).$$

Given the specification of the dynamics of a Boolean network T_{async} , the general method for determining all the maximal SCCs of its state transition graph $G(T_{\text{async}})$, consists in exhaustively analysing the graph. As already mentioned the number of nodes in $G(T)$ is $N = 2^n$, so that in order to make effective any procedure to compute $H(T_{\text{async}})$, we need polynomial bounds for:

1. k where $\mathcal{M}(T_{\text{async}}) = \{m_1, \dots, m_k\}$, that is the number of maximal SCCs,
2. $l = \max_{1 \leq i \leq k} k_i$ where $m_i = \{s_{i,1}, \dots, s_{i,k_i}\}$, that is the cardinality of the largest SCC,
3. d that is the length of the longest path between two SCCs; in particular, this measure is bounded by the diameter of the state transition graph $G(T_{\text{async}})$.

With the above notations, we have the following

Theorem 1. *The SAT-complexity of determining $H(T_{\text{async}})$ is bounded by*

$$o((k + \log_2 d) l d).$$

If the three quantities $k, l, d < n^c$ for some c , the procedure is effective with regards to the number of calls to the SAT solver; i.e., it is polynomial-time in the number of calls to the SAT solver and we denote this execution time as $o_{\text{SAT}}(n^{3c})$.

In the rest of the paper, we constructively show that there exists an algorithm which effectively finds the $H(T_{\text{async}})$ for those Boolean networks which satisfy complexity bounds in the hypothesis.

We tackle the problem of efficiently determining the hierarchical transition graph of an asynchronous network by visiting only nodes of the state transition graph which belong to the SCCs and paths connecting them. In order to obtain the optimal solution, this work relies on the integration of (i) the best known algorithm to determine SCCs, *i.e.*, the algorithm from Tarjan and (ii) the use of Boolean formulas for expressing any path which belongs to the graph. This condition enables the use of a SAT solver in order to find assignments to the variables which satisfies the path-formula. Moreover, the path-formula can be enriched with extra conditions on the represented states, avoiding that nodes of already discovered loops appear again on the path obtained by the SAT solver. This technique allows to overcome limitations of an exhaustive approach. It has indeed been successfully applied in the determination of the limit cycles of gene regulatory networks using a synchronous update. In the asynchronous case, the results are very limited since the structure of cycles as described by the SCCs of $G(T_{\text{async}})$ is much more complex.

3 Finding Limit Cycles in Synchronous Networks

The first ingredient of our work is the possibility of computing cycles in a state transition graph without exhaustively exploring it in its entirety. Indeed, for the gene regulatory networks of our interest that are composed of several dozens of genes, the corresponding state transition graph is so large that it is not even possible to store it in current digital memories.

To overcome such limitations, several approaches have been suggested, all sharing the idea to treat the updating function in a symbolic manner, like in [DT11, BGS06, ZYL+13]. To this end, there have been various proposals such as Binary Decision Diagrams [GMDC+09], Algebraic Decision Diagrams [BFG+93], Boolean Expressions and Logic Programming [HMMK13].

The approach suggested by Dubrova and Teslenko in [DT11] consisted in starting from the Boolean expressions of the components f_v of the dynamics and in defining a Boolean expression in the variables, representing relations between successive states in a path of the graph $G(T_{\text{sync}})$:

$$\text{StepExpression}[t] := \bigwedge_{i=1}^n s_i^t \leftrightarrow f_{v_i}(s_1^{t-1}, \dots, s_n^{t-1}). \quad (2)$$

Then, by iterating the expression for a number of steps from $t - k$ to t , the algorithm gets the expression corresponding to a Boolean Expression which any path in $G(T_{\text{sync}})$ has to satisfy:

$$\text{PathExpression}[t - k, t] := \bigwedge_{i=0}^{k-1} \text{StepExpression}[t - i].$$

Note that the number of variables involved in each formula depends on the number of the genes and the length of the path, namely, if we identify the size of the expression with the number of variables, we have

$$|\text{PathExpression}[t - k, t]| = n^k.$$

In Algorithm 1, we present the algorithm proposed by Dubrova. It consists in a while-loop which at each iteration performs a call to a SAT solver in order to test if a certain Boolean expression admits a solution. The Boolean expression is the conjunction of the `PathExpression` and a condition excluding that new paths have nodes which belong to already discovered cycles. If a solution exists, then the presence of a cycle is easily verified by testing for the presence of repeated states in the solution path provided by the SAT solver (`CheckPath` function). When the algorithm does not find any cycle in a path, then the `PathExpression` doubles the path length. This last step is very important since, when the path length becomes longer than the diameter of state transition graph and at the same time the algorithm has already visited any cycle in state transition graph, then the formula F becomes unsatisfiable and the exit condition of the loop is reached. From the viewpoint of the run-time complexity, this method performs a number of iterations which in the worst case is bounded by twice the logarithm of the diameter plus the number of cycles in $G(T_{\text{sync}})$. The number of iterations of the main cycle does not provide the complexity of the algorithm in the usual sense, since at each iteration we call the SAT solver which has an exponential complexity bound (on the size of the formula).

Thanks to this approach it is possible to establish limit cycles of synchronous GRNs consisting of a great number of genes. Our numerical experiments are in line with results reported in [DT11] and it is even possible to find limit cycles of a realistic network with 51 genes [PBC+10] in a matter of seconds, which would be impossible to achieve by using an exhaustive search algorithm.

In Table 1, we give an appreciation of how much better the symbolic approach is with respect to the exhaustive search: note that the number of nodes visited by our method (fourth column), is in the order of hundreds against the huge number of nodes appearing in the state transition graph ($N = 2^n$) even for networks consisting of many nodes (*e.g.*, last rows of the table).

Note that the maximal length of the paths does not need to be known in advance. In fact, although it depends on the network diameter, in practice it is found dynamically: the algorithm ends when, by doubling the length of the paths, this number exceeds the diameter and there is no path which can satisfy the formula F , since the algorithm has already found all the cycles.

Algorithm 1. Dubrova-Teslenko Algorithm to find limit cycles of $G(T_{\text{sync}})$ starting from the Boolean expression of T_{sync} .

Require: Boolean expression `PathExpression` which is satisfied by any path in the dynamics T_{sync} ; a global *stack* data structure representing the intermediate state of the calculation of the HTG.

```

1: function Cycles( $T$ )
2:   Initialise
3:    $path\_length := 1$ 
4:    $F := \text{PathExpression}(-path\_length, 0)$ 
5:   while Satisfiability( $F$ ) do
6:      $(c_{-path\_length}, \dots, c_0) := \text{SAT}(F)$ 
7:     if CheckPath( $(c_{-path\_length}, \dots, c_0)$ ) then
8:        $c_j$  minimal state forming the loop
9:        $Attractors(s_0) := Attractors(s_0) \wedge (s_0 \leftrightarrow c_j)$ 
10:       $F := F \wedge \neg Attractors(s_0)$ 
11:     end if
12:     if attractor_is_found then
13:       attractor_is_found := false
14:     else
15:        $F := \text{PathExpression}(-2path\_length, 0)$ 
16:        $path\_length := 2path\_length$ 
17:     end if
18:   end while
19: end function

```

Table 1. Statistics of the runs of our implementation of Dubrova Algorithm on several literature GRNs using the synchronous updating dynamics.

n	GRN name	# Limit cycles	Visited nodes	Paths (# SAT calls)	Max path length	Reference	Time
10	Fission yeast	13×1	28	16	8	[GMDC+09]	0.45
10	Mammalian cell	$1 \times 7, 1 \times 1$	29	6	16	[DT11]	0.25
12	Budding yeast	7×1	52	12	16	[DT11]	1.11
15	Arabidopsis thaliana	10×1	45	14	16	[DT11]	1.74
23	T-helper cell	3×1	42	7	16	[DT11]	0.29
40	T-helper cell receptor	$1 \times 6, 8 \times 1$	136	14	32	[DT11]	3.08
51	Th1/Th2 Switch	$1 \times 3, 3 \times 1$	97	10	64	[PBC+10]	6.06
52	Drosophila megalonoster	7×1	172	13	32	[DT11]	6.10
54	MAPK	$7 \times 8, 2 \times 7, 4 \times 4, 1 \times 2, 3 \times 1$	295	22	32	[GCBP+13]	62.00

4 The Algorithm for the Asynchronous Case

What described in the above section leads us to the conclusion that at least in the synchronous case, even when the number of genes produces a large state transition graph and its size makes any tentative to determine limit cycles

unreasonable, we have an effective tool which helps addressing (and solving indeed) the problem of finding limit cycles. In the asynchronous case, we have a different formula describing a single step, that is, instead of Eq. 2 we have to use the following one:

$$\text{StepExpression}[t] := \bigvee_{i=1}^n \left(s_i^t \leftrightarrow f_{v_i}(s_1^{t-1}, \dots, s_n^{t-1}) \wedge \bigwedge_{\substack{j=1 \\ j \neq i}}^n s_j^t \leftrightarrow s_j^{t-1} \right). \quad (3)$$

As it is evident, this formula has a greater logical complexity with respect to the one corresponding to the synchronous case, that is Eq. 2; a fact which reflects in the more intricate nature of the $G(T_{\text{async}})$.

Algorithm 1 works on the assumption that T_{sync} behaves as a deterministic function and only one possible transition can occur after a state; in this way a cycle is certainly found if the path is long enough. Moreover, once a path reaches a loop, it never leaves it (because of the uniqueness of the successor state).

In the asynchronous case, for any state we possibly have n successors (one for each gene, *i.e.*, component, *i.e.*, updating rule). The case in Fig. 1 is more complicated because *non terminal cycles* exist, making the problem of determining the structure of $G(T_{\text{async}})$ more similar to the identification of SCCs in a directed graph.

The strongly connected components of a directed graph can be found using a variant of the depth-first search (the method was originally devised by R.E. Tarjan in 1972, as stated above). Since it is based on the depth-first search (DFS), it runs in time proportional to $|V| + |E|$. It is worth to mention that before Tarjan, no linear time algorithm (in the number $|E|$) was known for this problem. A straightforward approach to the same problem is to follow a path-based algorithm as initially proposed by Purdom [Pur70] and Munro [Mun71] for strong components, later deeply analysed by Gabow [Gab00]. As a consequence of the structure of Dubrova’s algorithm which at each iteration generates one path, the “path oriented” approach to finding maximal SCCs in the state transition graph is the more appropriate choice.

We now describe the steps undertaken to design the algorithm which combines those of Dubrova (Algorithm 1) and Tarjan without affecting their complexity bounds. The structure of Algorithm 1 is unchanged, since the algorithm makes calls to the SAT solver to find an assignment to the variables which appears in the expression specifying the path of length *path.length* in the asynchronous dynamics. Note that we do not change the definition of $\text{PathExpression}[t-k, t]$ but $\text{StepExpression}[t]$ has been replaced with the one given in Eq. 3. In this case, because of the different dynamics, we have to consider that cycles can appear also in non-terminal SCCs thus, by running Algorithm 1, we have to detect cycles as part of SCCs in $G(T_{\text{async}})$. In order to do so, we interleave the execution of Algorithm 1 with the path-oriented Tarjan algorithm to discover SCCs. Instead of calling the function `CheckPath` which tests the presence of a loop in the path (*i.e.*, line 7 of Algorithm 1) a more subtle implementation of this test is required: a new function `CheckPath` is given as Algorithm 2, it can be seen

as the partial evaluation of the strongly connected components determination algorithm at each iteration and is our main contribution to this work.

Unfortunately, in Tarjan’s Algorithm the depth first search visit is performed by exploring the graph in a precise order, that is, by following all outgoing edges of the encountered nodes. In our case, however, we have to follow the sequence of nodes in the path provided by the SAT solver. By imposing the $\neg \text{Attractors}$ condition in the formula F , we are sure that the path provided by the SAT solver does not contain any node belonging to previously discovered SCCs, nevertheless a node already encountered in a transient path can appear again in new paths.

Algorithm 2. Tarjan DFS specialised to path processing

Require: the global *stack* data structure defined in Algorithm 1

```

1: function CheckPath( $v :: path$ )
2:   if  $path \neq emptypath$  then
3:      $w :: path' := path$ 
4:     if  $w$  is marked then
5:       ProcessBacklink( $v, w$ )
6:     else
7:       Push( $w$ )
8:       Mark( $w$ )
9:     end if
10:    CheckPath( $w :: path'$ )
11:  end if
12: end function

```

A difference with a generic depth first search algorithm is that a backtracking function which closes the visit for a given node is not required since it is not possible to say if a node will not be found again in next path to be analysed, until we reach the end. Therefore, the visit of the sub-graph of paths starting in a given node is never closed. Actually, when a node is absorbed in a SCC, then the visit of the sub-graph is completed and the component is represented by the node in the hierarchical transition graph which corresponds to its SCC.

What remains to be analysed are the functions used in the `ProcessBacklink`, which is the main novelty of our work. The structure we use is a stack; one stack for each thread, see Fig. 3. The function `Push` performs the push of a node on the stack. New threads are created when the solution provided from the SAT solver contains a node that has never been encountered before in first position. A node found more than once has multiple outgoing edges in its stack and therefore the stack is a tree. In order to detect loops, we fix an ordering between threads and we add pointers from nodes belonging to threads that are in ‘higher’ positions to nodes which are in ‘lower’ positions. When this ordering is broken (*i.e.*, `ThreadPrecedes` is false), because the edge we would like to add connects a node in a lower thread to a higher one, then a merge operation is issued (`MoveSubtree`) and the subtree in higher position is moved under the node in lower position. We test for the presence of a loop in the branch of the tree

Algorithm 3. Function `ProcessBacklink` specialised for the path oriented version of Tarjan.

Require: the global *stack* data structure defined in Algorithm 1

```

1: function ProcessBacklink( $v, w$ )
2:   if TestLoop( $v, w$ ) then
3:     CollapsePath( $w, v$ )
4:   else
5:     if ThreadPrecedes( $w, v$ ) then
6:       PushPointer( $v, w$ )
7:     else
8:       MoveSubtree( $w, v$ )
9:     end if
10:  end if
11: end function

```

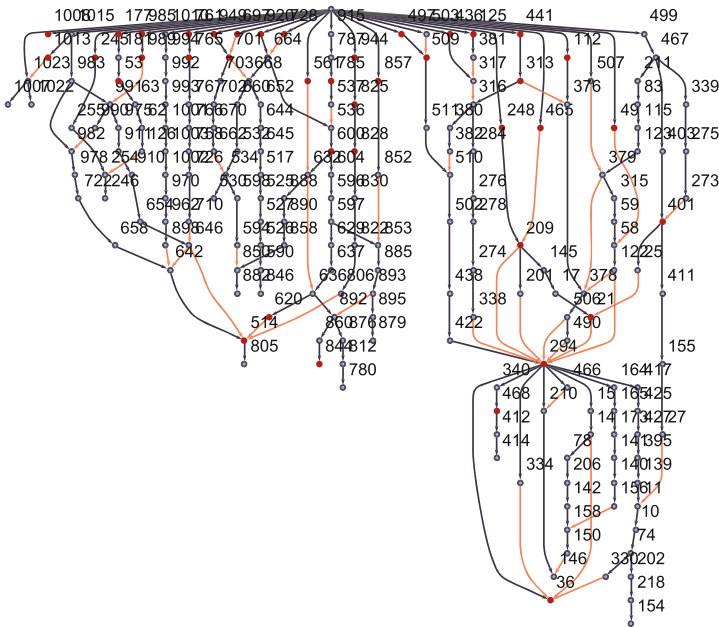


Fig. 3. Example of the data structure of the multithreaded stack (red-links represent backlinks, red nodes represent SCCs collapsed in that node). (Color figure online)

which contains the node w . If a loop is found then we collapse the path to a single *canonical node* representing an entire component by using `CollapsePath`.

5 Testing on Random Networks

We report about several runs performed on random Boolean networks. These are specified as directed graphs whose links represent either activations or inhibitions

Table 2. Runtime best and worst case out of 20 randomly generated GRNs with both synchronous and asynchronous dynamics. See symbol legend in the paragraph text. Missing data correspond to cases exceeding a bound in the execution time. When runtime exceeds a given time-out (heuristic dependent on the number of genes) they are shown among parentheses.

n		v	$v/2^n$	paths (# SAT calls)	max length of paths	density (K/N)	inh/(act+inh)	time
5	S	8	.25	3	4	0.36	0.11	0.09
		15	.47	6	8	0.32	0.01	0.16
	A	16	.50	6	16	0.48	0.08	1.28
		32	1.00	19	8	0.48	0.08	5.24
7	S	11	.09	4	8	0.31	0.07	0.10
		41	.32	11	8	0.35	0.29	0.30
	A	57	.45	12	16	0.29	0.00	4.39
		128	1.00	62	16	0.14	0.15	189.89
8	S	5	.02	3	4	0.19	0.17	0.10
		31	.12	8	16	0.36	0.30	0.25
	A	144	.56	12	32	0.27	0.18	25.04
		256	1.00	125	8	0.14	0.22	486.15
10	S	17	.17	7	8	0.1	0.2	0.18
		153	.15	20	16	0.17	0.18	3.81
	A	210	.21	10	64	0.2	0.1	34.99
		1024	1.00	485	16	0.23	0.48	7703.87
11	S	14	.01	5	8	0.17	0.01	0.17
		36	.02	10	8	0.17	0.01	0.53
	A	286	.14	17	64	0.19	0.0	100.22
		2048	1.00	572	16	0.15	0.11	27188.90
12	S	14	3×10^{-3}	5	8	0.17	0.01	0.17
		36	.01	10	8	0.17	0.01	0.53
	A	302	.07	13	64	0.23	0.18	56.6134
		3056	.75	300	64	0.17	0.02	49736.80
16	S	16	2×10^{-4}	4	8	0.13	0.01	0.16
		32	5×10^{-4}	7	16	0.13	0.01	0.36
	A	68	1×10^{-3}	7	32	0.14	0.09	14.07
100	S	18	1×10^{-29}	4	8	0.15	0.02	0.60
		367	3×10^{-28}	21	32	0.15	0.02	139.85
	A							
1000	S	32	3×10^{-300}	5	16	0.10	0.01	96.85
		63	6×10^{-300}	7	32	0.15	0.02	931.59
	A							

among genes. Two parameters control the generation of the graph: (i) the *density* parameter δ gives the probability that a gene influences another gene, namely that there is a link between them; (ii) the probability α that such a link is an *inhibition* (respectively $1 - \alpha$ for *activation*). What we report in Table 2 is the best and worst case in terms of performance computed on twenty independent runs for both synchronous and asynchronous updating rules. In the table S stays for synchronous update and A for asynchronous; p is the number of generated paths which coincides with the number of SAT-solver calls; m is the maximal length of paths (it relates to the diameter of the STG); δ is the density of the GRN (fraction of links w.r.t. the fully linked network); α is the fraction of inhibitors over the whole number of links in the GRN. The elapsed time t in seconds correlates to the number of *visited nodes* v ; the fraction of the whole STG that was necessary to explore in order to determine the HTG is $v/2^n$ as reported in 4th column.

Moreover whereas in the synchronous case there is a direct correlation between t and m , in the asynchronous case this relationship is inverted. Note that since the networks are drawn at random, the algorithm performs efficiently when the hypotheses of Theorem 1 are satisfied. For instance this is the case of the network with $n = 16$ for which $v/2^n$ equals to $585/2^{16} \simeq 10^{-3}$ which in fact terminates in about 136 s. An opposite case is that of $n = 10$ for which the algorithm explores the totality of nodes of the state transition graph ($937/2^{10} = 90\%$) which runs for more than one hour due to the fact that the state transition graph is made of many relatively small SCCs, a fact that translates to a smaller value for m .

Note that the time values shown in the last column of Table 2 derive from an implementation of the algorithm which does not conform to the criteria of high performance and should therefore be considered as an indication of the relationships among execution time and network characteristics as just discussed. For those cases when the execution time exceeds a heuristic threshold (value dependent on the number of genes), the time spent until that limit is shown among parentheses. In the largest case of $n = 10^3$ the huge size of the search space prevented the algorithm finding a single time for the execution of the asynchronous case.

6 Conclusion

We have presented a method that can efficiently determine the attractors of a gene regulatory network in the case of the asynchronous updating rule by combining the formulation of a dynamical system in terms of satisfiability problems with an efficient algorithm for determining the strongly connected components of a graph; resulting in the possibility to determining the hierarchical transition graph without the need to entirely exploring the state transition graph. The method presented here extends previous algorithms developed for the synchronous dynamics to the asynchronous case which is regarded as being more

realistic. The existence of the algorithm is a proof of Theorem 1 which summarises the link between the size of the hierarchical transition graph and the run-time.

A more detailed description of the implementation will be provided in a follow-up manuscript.

References

- [BCM+13] Bérenguier, D., Chaouiya, C., Monteiro, P.T., Naldi, A., Remy, E., Thieffry, D., Tichit, L.: Dynamical modeling and analysis of large cellular regulatory networks. *Chaos: Interdisc. J. Nonlinear Sci.* **23**(2), 025114 (2013)
- [BFG+93] Bahar, R.I., Frohm, E.A., Gaona, C.M., Hachtel, G.D., Macii, E., Pardo, A., Somenzi, F.: Algebraic decision diagrams and their applications. In: 1993 IEEE/ACM International Conference on Computer-Aided Design, ICCAD 1993, Digest of Technical Papers, pp. 188–191. IEEE (1993)
- [BGS06] Bloem, R., Gabow, H.N., Somenzi, F.: An algorithm for strongly connected component analysis in $n \log n$ symbolic steps. *Formal Methods Syst. Des.* **28**(1), 37–56 (2006)
- [DJ02] De Jong, H.: Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.* **9**(1), 67–103 (2002)
- [DT11] Dubrova, E., Teslenko, M.: A SAT-based algorithm for finding attractors in synchronous Boolean networks. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **8**(5), 1393–1399 (2011)
- [Gab00] Gabow, H.N.: Path-based depth-first search for strong and biconnected components. *Inf. Process. Lett.* **74**(3–4), 107–114 (2000)
- [GCBP+13] Grieco, L., Calzone, L., Bernard-Pierrot, I., Radvanyi, F., Kahn-Perlès, B., Thieffry, D.: Integrative modelling of the influence of mapk network on cancer cell fate decision. *PLoS Comput. Biol.* **9**(10), e1003286 (2013)
- [GMDC+09] Garg, A., Mohanram, K., Di Cara, A., De Micheli, G., Xenarios, I.: Modeling stochasticity and robustness in gene regulatory networks. *Bioinformatics* **25**(12), i101–i109 (2009)
- [HB97] Harvey, I., Bossomaier, T.: Time out of joint: attractors in asynchronous random Boolean networks. In: *Proceedings of the Fourth European Conference on Artificial Life*, pp. 67–75. MIT Press, Cambridge (1997)
- [HMMK13] Hopfensitz, M., Müssel, C., Maucher, M.: HA Kestler: attractors in Boolean networks: a tutorial. *Comput. Stat.* **28**(1), 19–36 (2013)
- [Kau93] Kauffman, S.A.: *The Origins of Order: Self-organization and Selection in Evolution*. Oxford University Press, Oxford (1993)
- [Mun71] Munro, I.: Efficient determination of the transitive closure of a directed graph. *Inf. Process. Lett.* **1**(2), 56–58 (1971)
- [PBC+10] Pedicini, M., Barrenäs, F., Clancy, T., Castiglione, F., Hovig, E., Kanduri, K., Santoni, D., Benson, M.: Combining network modeling and gene expression microarray analysis to explore the dynamics of Th1 and Th2 cell regulation. *PLoS Comput. Biol.* **6**(12), e1001032 (2010)
- [Pur70] Purdom, P.: A transitive closure algorithm. *BIT Numer. Math.* **10**(1), 76–94 (1970)

- [Tar72] Tarjan, R.: Depth-first search and linear graph algorithms. *SIAM J. Comput.* **1**(2), 146–160 (1972)
- [VCL12] Veliz-Cuba, A., Laubenbacher, R.: On the computation of fixed points in Boolean networks. *J. Appl. Math. Comput.* **39**(1–2), 145–153 (2012)
- [ZYL+13] Zheng, D., Yang, G., Li, X., Wang, Z., Liu, F., He, L.: An efficient algorithm for computing attractors of synchronous and asynchronous Boolean networks. *PLoS ONE* **8**(4), e60593 (2013)



The Impact of Self-loops in Random Boolean Network Dynamics: A Simulation Analysis

Sara Montagna^(✉), Michele Braccini^(✉), and Andrea Roli^(✉)

Department of Computer Science and Engineering, Campus of Cesena,
Alma Mater Studiorum, Università di Bologna, Cesena, Italy
{sara.montagna,m.braccini, andrea.roli}@unibo.it

Abstract. Random Boolean Networks (RBNs) are a popular and successful model of gene regulatory networks, especially for analysing emergent properties of cell dynamics. Since completely random networks are unrealistic, some work has been done to extend the original model with structural and functional properties observed in biological networks. Among recurring motifs identified by experimental studies, auto-regulation seems to play a significant role in gene regulatory networks. In this paper we present a model of auto-regulatory mechanisms by introducing self-loops in RBNs. Experiments are performed to analyse the impact of self-loops in the RBNs asymptotic behaviour. Results show that the number of attractors increases with the amount of self-loops, while their robustness and stability decrease.

1 Introduction

Boolean Networks (BNs) have been successfully used as gene regulatory network (GRN) models both for identifying generic properties of cell dynamics [4, 14, 22, 28] and for reproducing a specific (partial) genetic network reconstructed from biological data [8, 26]. When generic properties are sought, the typical approach consists in studying ensembles of boolean networks generated according to a given, biologically plausible, model, such as the one proposed by Kauffman [14]. In this model a Random Boolean Network (RBN) is initialised completely random both in the topology and in the functions, possibly defining the number of inputs each node has. Variants of this model have also been considered, for example by restricting the set of boolean functions to canalising ones, or by imposing a scale-free topology. These variants are inspired by biological plausibility and are often suggested by the identification of crucial properties and mechanisms observed in GRNs reconstructed from biological data.

The work presented in this paper is part of this research field. The long term goal is to identify basic mechanisms and common motifs of GRNs underlying fundamental cellular processes [24, 29] that can be modelled as structural and functional elementary bricks in BNs, thus making it possible to study generic properties of cell dynamics by means of ensembles of more realistic BNs models.

Furthermore, this repertoire of bricks may be used inside algorithms for the automatic generation of BNs endowed with specific dynamical properties [6, 7]. Such networks may also be exploited for designing and controlling the behaviour of artificial entities [9].

As a first step, in this paper we analyse the role and the impact of self-loops, which abounds in biological genetic networks, in RBNs dynamics. Indeed, within a GRN, a self-loop models the property of a gene producing some chemical substances that contribute at the regulation of its own gene. In particular we focus here on positive self-loops, whose effect is to maintaining the activation state of the gene.

We performed different simulation experiments where a RBN created completely random is incrementally modified introducing one self-loop at a time until every node has one. Different configurations, in terms of network topology and functions, are evaluated. Results show that self-loops have a crucial role in RBN asymptotic states and the same trend is observed, independently from the specific configuration: as the percentage of self-loops increases, the number of attractors increases exponentially while their stability decreases almost linearly.

2 Motivation and Goal

BN is one of the formalisms adopted to model GRNs. The most used model defines a RBN of n nodes according to the following rules: each node has exactly k inputs, randomly chosen among the other nodes; boolean functions are assigned to each node on the basis of the 2^k truth table entries, in which the entry is set to 1 with probability p (the *bias*). A node of a RBN models a gene whose expression is regulated by its k input genes. Boolean functions model the regulation type such as, for instance, activation or inhibition. Formally, a BN is defined by a directed graph of N nodes, each associated to a Boolean variable x_i , $i = 1, \dots, N$, and a Boolean function $f_i(x_{i_1}, \dots, x_{i_k})$ with $\{x_{i_1}, \dots, x_{i_k}\} \not\ni x_i$, *i.e.*, self-loops are not allowed; k is the number of inputs of node i . The arguments of the Boolean function f_i are the values of the nodes whose outgoing arcs are connected to node i . The state of the system at time t , $t \in N$, is defined by the array of the N Boolean variable values at time t : $s(t) \equiv (x_{1(t)}, \dots, x_{N(t)})$.

Literature suggests that RBNs are particularly effective as an abstract model of GRNs for reproducing the most relevant features of experimentally observed phenomena. In particular, much work has been done to compare the RBNs asymptotic states to GRNs evolution. A notable example is the work proposed by Kauffman [14] where RBNs are used as a model that describes the dynamics of cellular differentiation. There, attractors of RBNs are identified with cell types, since each state corresponds to the dynamic activation of a subset of nodes, *i.e.*, of a subset of genes that are the markers of a specific cell type or differentiation stage.

From there, research efforts extended the basic model to include features and mechanisms that are significant in the biological systems. Extensions investigate for instance the role of noise in the stability of the asymptotic states [12, 28], since

it plays a crucial role in cellular regulatory networks [17]. Other works [10, 15] discuss whether boolean rules, if selected randomly among all the possible boolean functions of k inputs, are an acceptable approximation to model gene regulation. Their conclusion states that canalising rules reproduce experimental observation more accurately. Finally, the distribution of GRNs is analysed, observing that they mainly exhibit a scale-free topology [2], and simulation experiments with scale-free RBNs are performed, suggesting that not pure random neither scale-free are likely the best approximation for GRN topology and that further studies are worthy [3, 21, 23].

Given these premises, we believe that analysing common motifs and basic mechanisms of GRNs, and identifying the best solution for modelling them as structural and functional elementary bricks in RBNs, would support a more aware analysis and understanding of the emergent dynamics obtained with the simulation [1, 29]. The research presented in this paper grounds on this vision. Our long term goal is to build a catalogue of bricks—similarly from the BioBricksTM idea of Synthetic Biology [25]—whose function and role inside a GRN is known. The expected impact of this bricks catalogue is twofold. On one side the analysis of GRNs dynamics via simulation will provide an in depth clue on the link between the function of the parts and the emergent behaviour observed. On the other side, the engineering and design of RBNs with specific behaviours will then be possible by composing known bricks.

We present in this paper a first step towards this challenging result. The network motif under study here is *self-loop*, also known as auto-regulation mechanism, *i.e.*, the gene regulation motif where a transcription factor regulates the transcription of its own gene. Self-loops abound in biological genetic networks [11]. In this paper we focus on positive self-loops responsible for the up-regulation of their own genes. This mechanism is particularly evident in the differentiation process, where cells, from a stem state, choose a fate towards specific specialised cells. For example, from the *Drosophila* GRN shown in Fig. 8 of [18], all the four main genes responsible for the patterning of gap genes expression during embryo development are involved in autocatalytic reactions.

To the best of our knowledge, the impact of self-loops has only preliminarily been studied in RBNs. A first work is presented in [20], where the relation between the sign of the regulation (positive or negative) and the robustness of the network is investigated.

3 Methods

The impact of the introduction of self-loops into a RBN has been studied through simulation. In particular the goal was to observe how the asymptotic behaviour of the network, namely the number of attractors and their stability, changes as a function of the fraction of self-loops added.

As in the RBN model introduced by Kauffman [13], we suppose that one node in the RBN corresponds to one gene in the GRN. For simulation purposes, we modified a randomly generated boolean network in different ways; we have:

AUGM-RND: added a self-loop and extended the truth table randomly (with the same bias used for generating the original RBN);

AUGM-OR: added a self-loop and changed the node boolean function into an OR between the node value and the previous function;

CONST-RND: removed an incoming link and replaced the input with a self-loop, without changing the node boolean function;

CONST-OR: removed an incoming link and replaced the input with a self-loop, and changed the node boolean function into an OR between the node value and the previous function.

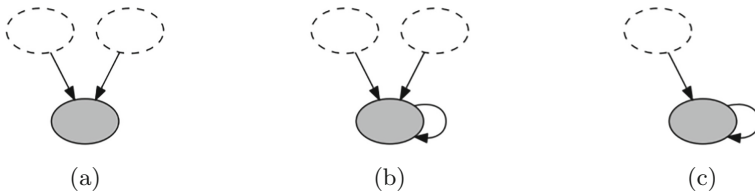


Fig. 1. a: Typical generic node generated following the original RBN model with $k = 2$, self-loops are not allowed. b: Example of a node modified for experiments with a self-loop added. c: Example of a node modified for experiments with an incoming link removal and self-loop addition. Outcoming arcs are not drawn since both the RBN model and our models do not impose constraints on them.

In Fig. 1 we elucidate how a single network node is modified to obtain the experiments configurations previously mentioned. We thus explore the role of self-loops in both the cases of maintaining a random boolean function and of adopting a canalising function (OR). The choice to explore canalising functions is motivated by the focus on self-loops with self-activating effect. Indeed, according to the role they have in biological networks, a positive self-loop should model the property of a gene producing some chemical substances that contribute maintaining the activation state of that gene. This means, within a RBN, that the function should keep the node value. It is worth mentioning that, since within a RBN we do not associate 0 with gene off and 1 with gene on, but we are interested in maintenance and transition of states, using an OR function or an AND function is conceptually the same.

For each of these experiments, self-loops are introduced incrementally to the original RBN. In this way we have a fraction of self-loops varying from 0 to 1 (no node has a self-loop – all nodes have a self-loop) and we are able to observe how the behaviour of the network is modified step-by-step.

In all experiments, each RBN is simulated following a synchronous dynamics update scheme—*i.e.*, nodes update their states at the same instant—and with

deterministic functions. Since the state space is finite, the BN after a transient eventually reaches a *fixed point* or a *cyclic attractor*; these are the only achievable asymptotic states in this setting.

Statistics are taken across 50 different RBNs with $n = 20$ nodes. Initial RBNs are created with $k = 2$ and function bias p equal to 0.5. The value of the k parameter is chosen for its biological plausibility [13, 16, 22]. The ensemble of RBNs having these bias and k values are in critical dynamic regime [5]. We sampled from this networks ensemble because, statistically, they exhibit robustness and adaptiveness similar to real genetic regulatory networks [27, 28]. In the experiments, we explored all the possible initial states (2^{20}) of the RBN, to obtain the whole attractors landscape. Since we did not want the results of comparisons between models to be affected by the variance of the network dynamic regime, we set an exact bias to initial RBNs (before any modification). To do this, we computed a random permutation of a vector with length equal to the sum of all Boolean functions entries ($2^k * n$) with half of the values to 1 and the remaining to 0; we take a portion of this vector to populate the truth table of a node.

To estimate the dynamic robustness of a network we introduced noise, modelled by a random flip of a randomly chosen node (logic negation of a node state). We have flipped each node of each state of each attractor in order to compute a matrix, called Attractor Transition Matrix (ATM), that summarises the transition probabilities between attractors; the procedure is described in [19, 28]. In particular, to compute our statistics measuring the network robustness, we examine the main diagonal of the ATM: each diagonal entry give us the estimate probability of returning in the same attractor as a result of a random node perturbation.

4 Results

Results reveal that self-loops massively affect the number of attractors and their robustness. Figures 2, 3, 4 and 5 show how the average number of attractors and the probability of returning to an attractor vary as a function of the fraction of self-loops. In particular, on the left side of Figs. 2, 3, 4 and 5, each point corresponds to the average number of attractors obtained across the 50 networks with a particular fraction of self-loops. On the right side, we have the robustness trend as a function of the fraction of self-loops; each boxplot represents the distribution of the ATM main diagonal values from all 50 BNs.

Generally speaking, we can observe that the number of attractors is higher in the networks with self-loops. It grows quasi-exponentially, thus the effect observed is gradually more evident with increasing number of self-loops. In particular under around 30% of self-loops, the number of attractors slowly grows, not impacting too much the network dynamics, while afterwards it sharply increases, strongly reconfiguring the attractor landscape. At the same time, attractors' robustness tends to be smaller than in classical RBNs. This result is quite intuitive: since the number of attractors is significantly higher, the size of the basins

AUGM-RND

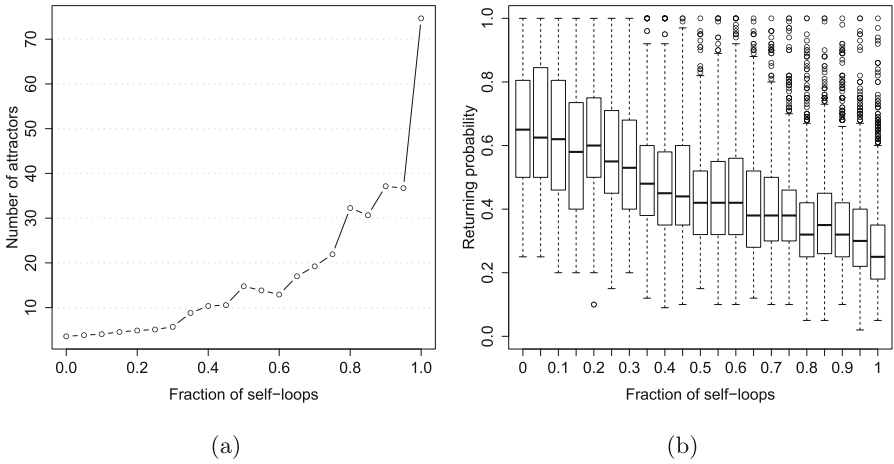


Fig. 2. Average number of attractors (*i.e.*, both cycles and fixed points) as a function of the fraction of self-loops added in RBNs originally with $k = 2$ (a). Distribution of the probabilities of returning to an attractator after one node flip (b).

AUGM-OR

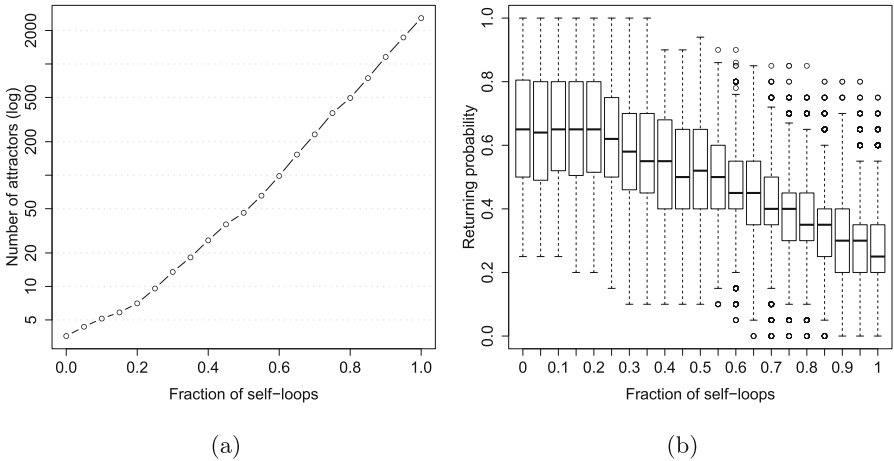


Fig. 3. Average number of attractors (*i.e.*, both cycles and fixed points) as a function of the fraction of self-loops added with an OR function in RBNs originally with $k = 2$ (a). Distribution of the probabilities of returning to an attractator after one node flip (b).

CONST-RND

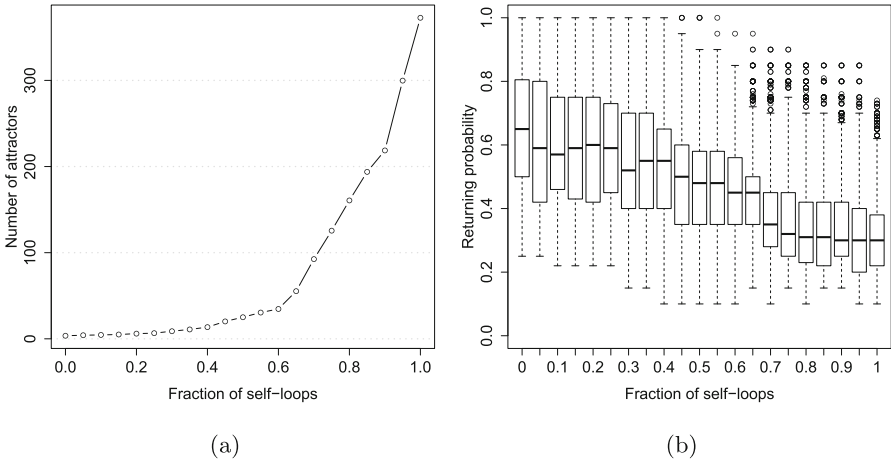


Fig. 4. Average number of attractors (*i.e.*, both cycles and fixed points) as a function of the fraction of self-loops added in RBNs originally with $k = 2$ (self-loops are introduced by rewiring a randomly chosen input) (a). Distribution of the probabilities of returning to an attractor after one node flip (b).

CONST-OR

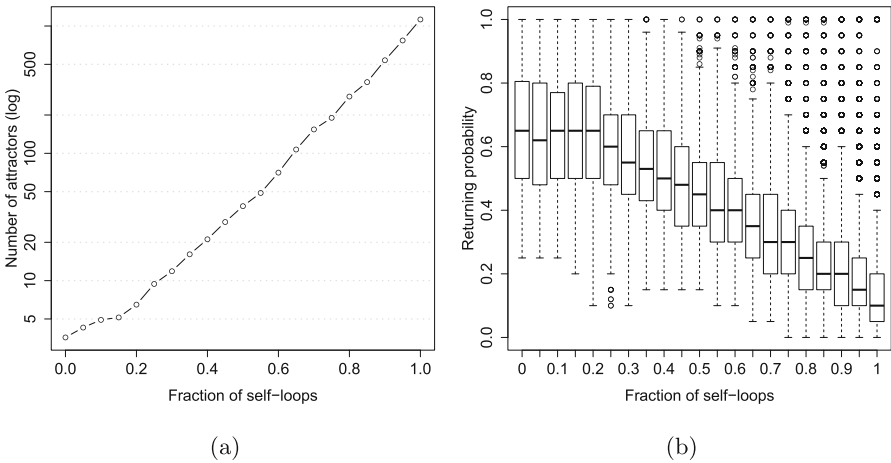


Fig. 5. Average number of attractors (*i.e.*, both cycles and fixed points) as a function of the fraction of self-loops added in RBNs with $k = 2$ (self-loops are introduced by rewiring a randomly chosen input and substituting the boolean function with an OR) (a). Distribution of the probabilities of returning to an attractor after one node flip (b).

of attraction should on average be smaller, thus making less likely to return to the same attractor after a flip, while easier to move in an other attractor. Moreover, the impact is even more striking when boolean functions are changed into an OR between the previous function and the node value involved in the self-loop.

Even though we have found that the results are qualitatively the same for all the four variants we considered for introducing self-loops (as discussed in Sect. 3), in the following we detail the main differences we observed.

AUGM-RND: the number of attractors varies quasi-exponentially until around 70 attractors for networks with a self-loop in each node. Conversely the median value of the returning probability decreases from around 0.6 to 0.3. However the distribution of the ATM main diagonal values is widely distributed between the extreme values of the range;

AUGM-OR: the peculiar characteristic of this experiment is that the number of attractors raises exponentially until around 2000 attractors. We used a logarithmic scale to zoom the plot to just a few nodes with self-loops; we motivate this significative difference in the attractors number, with respect to the previous configuration, noting that the OR function increases the probability that one node is in the 1 state, and it assures that it remains at that value;

CONST-RND: the number of attractors varies approximately exponentially until around 400 attractors, *i.e.*, more than the max number of attractors we have in experiments where we add a self-loop;

CONST-OR: the number of attractors varies exponentially until around 600 attractors. Peculiar in this setting is the median value of the returning probability graph where the probabilities of returning to an attractor, after one node flip, decreases until around 0.1, which is the lowest value we have in all the settings we considered.

4.1 RBNs with OR Functions

A question may be asked as to what extent the observed results depend on OR functions rather than self-loops. To address this question we first observe that in a BN with random topology and all OR functions it is very likely to have two attractors corresponding to two fixed points $S_0 = (00\dots 0)$ and $S_1 = (11\dots 1)$, characterised by a basin of attraction of 1 and $2^n - 1$, respectively. Therefore, in the limit case the number of attractors decreases—instead of increasing as in the case with self-loops—and so we expect experimentally. Results of—statistics over 50—experiments are summarised in Fig. 6.

We observed that the average number of attractors tends to decrease until almost 80% of OR functions within the network. This result is consistent with literature findings: from theoretical results is known that canalising functions

ONLY-OR

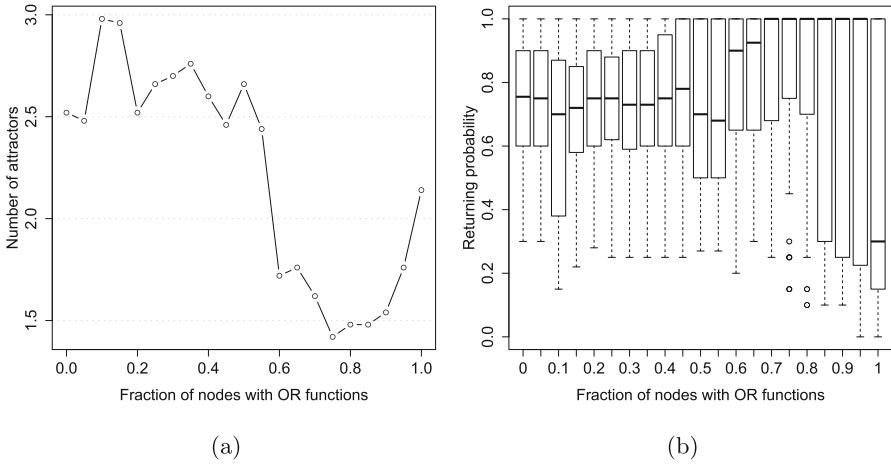


Fig. 6. Average number of attractors (*i.e.*, both cycles and fixed points) as a function of the number of nodes with an OR function (a). Distribution of the probabilities of returning to an attractor after one node flip (b).

move the RBN with $k = 2$ from a critical dynamic regime towards an ordered regime where we observed that the mean number of attractors is one. Thereafter, we observe a final increasing trend. We conjecture that it is due to the growing prevalence of network with two fixed points. In particular with all OR functions we measure an average number of attractors slightly bigger than two. We think that this result is owing to network topology that prevents the signal to be propagated to the whole network.

5 Discussion and Conclusion

If we want to stay close to the Kauffman interpretation of attractors, during the process of differentiation a RBN evolves and passes through different attractors that represent different cellular states, from stem cells to terminally differentiated cells. In this vision, the number of attractors models cellular diversity, while attractor stability models how strong must a signal be to move from one cell type to another. A tight balance between diversity and robustness ensures the perfect homeostasis known in multicellular organisms.

By analysing the impact of positive self-loops in RBN attractor landscape, we observed that they have an important role in network dynamics, and particularly on the number and stability of attractors. On one side they bring diversification, on the other side they seem to be responsible for instability. An operating point, where the balance is perfect as in biological world, is worth to be found.

More than that, biological research identified, for each differentiation state, a set of markers that characterise and identify the differentiation state. In RBNs, within each attractor, only a subset of nodes maintains its state (on/off). We here speculate that these nodes can model the concept of markers—an in-depth analysis of this claim is devoted to future work. In the model presented in this paper, self-loops are the mechanism that contributes maintaining the node state. This is particularly true considering those networks where self-loops are introduced with the OR function. There, their role is exactly to keep the local stability on a subset of nodes, representing the marker genes. If the network is in the operating point, “some self-loops but not too many”, they have the crucial role to cause diversification, *i.e.*, different cell types, without harming attractors stability, *i.e.*, cell type robustness.

Finding this operating point is thus crucial. However not trivial, especially because homeostasis in multicellular organisms is the result of a number of different mechanisms. This means that, including in the model also other phenomena and bricks, can change the dynamic described in this paper. In particular we draw the reader attention to auto-inhibitory processes, epigenetics and cell-to-cell interactions. Auto-inhibition negatively regulate gene expression. Epigenetics affects gene expression by changing the chromatin accessibility. As a consequence, cells with the same set of genes respond differently to the same signal. Cellular interactions influence the intracellular GRN dynamic by means of signals crosscutting cell membranes and whose effect is typically to activate or inhibit the expression of the target gene.

We conclude that results shown in this paper suggest to study the advantage of having self-loops in genetic networks during differentiation processes. However, further investigation is necessary to provide a more complete analysis and understanding of the results observed in this paper, supporting our findings with theoretical verification and estimation. Moreover, future work will be devoted to investigate what mechanisms counterbalance the effect of self-loops on attractor robustness, which is, as discussed previously, a fundamental property for modelling cell dynamics.

Finally, as mentioned in the Introduction section, we can think that this catalogue of bricks we are building, and whose functions we are analysing, are elementary building blocks that can be combined to face the reverse engineering problem of reconstructing real GRNs or designing GRN model with desired dynamics properties for artificial purposes. In addition, this approach can give us insights of the evolutionary processes that biological GRNs have undergone.

References





1. Ahnert, S.E., Fink, T.M.A.: Form and function in gene regulatory networks: the structure of network motifs determines fundamental properties of their dynamical state space. *J. Roy. Soc. Interface* **13**(120), 278–289 (2016)
2. Albert, R.: Scale-free networks in cell biology. *J. Cell Sci.* **118**(Pt 21), 4947–4957 (2005). <https://doi.org/10.1242/jcs.02714>

3. Aldana, M.: Boolean dynamics of networks with scale-free topology. *Phys. D Non-linear Phenom.* **185**(1), 45–66 (2003)
4. Balleza, E., Alvarez-Buylla, E.R., Chaos, A., Kauffman, S., Shmulevich, I., Aldana, M.: Critical dynamics in genetic regulatory networks: examples from four kingdoms. *PLoS One* **3**(6), e2456 (2008)
5. Bastolla, U., Parisi, G.: A numerical study of the critical line of Kauffman networks. *J. Theor. Biol.* **187**(1), 117–133 (1997)
6. Benedettini, S., Roli, A., Serra, R., Villani, M.: Automatic design of boolean networks for modelling cell differentiation. In: Cagnoni, S., Mirolli, M., Villani, M. (eds.) *Evolution, Complexity and Artificial Life*, pp. 77–89. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-37577-4_5
7. Braccini, M., Roli, A., Villani, M., Serra, R.: Automatic design of boolean networks for cell differentiation. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016. CCIS*, vol. 708, pp. 91–102. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_8
8. Chaos, Á., Aldana, M., Espinosa-Soto, C., de León, B.G.P., Arroyo, A.G., Alvarez-Buylla, E.R.: From genes to flower patterns and evolution: dynamic models of gene regulatory networks. *J. Plant Growth Regul.* **25**(4), 278–289 (2006)
9. Francesca, G., Brambilla, M., Brutschy, A., Trianni, V., Birattari, M.: Automode: a novel approach to the automatic design of control software for robot swarms. *Swarm Intell.* **8**(2), 89–112 (2014)
10. Harris, S.E., Sawhill, B.K., Wuensche, A., Kauffman, S.: A model of transcriptional regulatory networks based on biases in the observed regulation rules. *Complexity* **7**(4), 23–40 (2002)
11. Hermsen, R., Ursem, B., ten Wolde, P.R.: Combinatorial gene regulation using auto-regulation. *PLoS Comput. Biol.* **6**(6), 1–13 (2010). <https://doi.org/10.1371/journal.pcbi.1000813>
12. Hoffmann, M., Chang, H.H., Huang, S., Ingber, D.E., Loeffler, M., Galle, J.: Noise-driven stem cell and progenitor population dynamics. *PLoS One* **3**(8), 1–10 (2008). <https://doi.org/10.1371/journal.pone.0002922>
13. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.* **22**(3), 437–467 (1969)
14. Kauffman, S.A.: *The origins of order*. Oxford University Press, Oxford (1993)
15. Kauffman, S., Peterson, C., Samuelsson, B., Troein, C.: Random boolean network models and the yeast transcriptional network. *Proc. Nat. Acad. Sci.* **100**(25), 14796–14799 (2003)
16. Kauffman, S.A.: Homeostasis and differentiation in random genetic control networks. *Nature* **224**(5215), 177–178 (1969). <http://www.nature.com/doi/10.1038/224177a0>
17. McAdams, H., Arkin, A.: Stochastic mechanisms in gene expression. *Proc. Nat. Acad. Sci.* **94**(3), 814–819 (1997). <http://www.pnas.org/content/94/3/814.abstract>
18. Montagna, S., Viroli, M., Roli, A.: A framework supporting multi-compartment stochastic simulation and parameter optimisation for investigating biological system development. *Simul. Trans. Soc. Model. Simul. Int.* **91**, 666–685 (2015)
19. Paroni, A., Graudenzi, A., Caravagna, G., Damiani, C., Mauri, G., Antoniotti, M.: CABeRNET: a cytoscape app for augmented boolean models of gene regulatory networks. *BMC Bioinf.* **17**, 64–75 (2016)
20. Pinho, R., Garcia, V., Irimia, M., Feldman, M.W.: Stability depends on positive autoregulation in boolean gene regulatory networks. *PLoS Comput. Biol.* **10**(11), 1–14 (2014). <https://doi.org/10.1371/journal.pcbi.1003916>

21. Serra, R., Villani, M., Graudenzi, A., Colacci, A., Kauffman, S.A.: The simulation of gene knock-out in scale-free random boolean models of genetic networks. *Netw. Heterog. Media* **2**(3), 333–343 (2008)
22. Serra, R., Villani, M., Semeria, A.: Genetic network models and statistical properties of gene expression data in knock-out experiments. *J. Theor. Biol.* **227**, 149–157 (2004)
23. Serra, R., Villani, M., Agostini, L.: On the dynamics of random boolean networks with scale-free outgoing connections. *Phys. A: Stat. Mech. Appl.* **339**(3–4), 665–673 (2004). <http://www.sciencedirect.com/science/article/B6TVG-4C477JP-1/2/f6e8e45217874ad364008f770689a964>
24. Shen-Orr, S.S., Milo, R., Mangan, S., Alon, U.: Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nat. Genet.* **31**(1), 64–68 (2002)
25. Shetty, R.P., Endy, D., Knight, T.F.: Engineering biobrick vectors from biobrick parts. *J. Biol. Eng.* **2**(1), 5 (2008). <https://doi.org/10.1186/1754-1611-2-5>
26. Shmulevich, I., Dougherty, E., Kim, S., Zhang, W.: Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinf.* **18**(2), 261–274 (2002)
27. Shmulevich, I., Kauffman, S.A., Aldana, M.: Eukaryotic cells are dynamically ordered or critical but not chaotic. *Proc. Nat. Acad. Sci. U.S.A.* **102**(38), 13439–13444 (2005). <http://www.ncbi.nlm.nih.gov/pubmed/16155121%5Cnwww.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC1224670>
28. Villani, M., Barbieri, A., Serra, R.: A dynamical model of genetic networks for cell differentiation. *PLoS One* **6**(3), e17703 (2011)
29. Yeager-Lotem, E., Sattath, S., Kashtan, N., Itzkovitz, S., Milo, R., Pinter, R.Y., Alon, U., Margalit, H.: Network motifs in integrated cellular networks of transcription protein interaction. *Proc. Nat. Acad. Sci. U.S.A.* **101**(16), 5934–5939 (2004). <http://www.pnas.org/content/101/16/5934.abstract>



A Comparison Between Threshold Ergodic Sets and Stochastic Simulation of Boolean Networks for Modelling Cell Differentiation

Michele Braccini¹ , Andrea Roli¹ , Marco Villani^{2,3} ,
and Roberto Serra^{2,3} 

¹ Department of Computer Science and Engineering, Alma Mater Studiorum,
Università di Bologna, Bologna, Italy

m.braccini@unibo.it

² Department of Physics, Informatics and Mathematics,
Università di Modena e Reggio Emilia, Modena, Italy

³ European Centre for Living Technology, Venice, Italy

Abstract. Recently a cell differentiation model based on noisy random Boolean networks has been proposed. This mathematical model is able to describe in an elegant way the most relevant features of cell differentiation. Noise plays a key role in this model; the different stages of the differentiation process are emergent dynamical configurations deriving from the control of the intracellular noise level. In this work we compare two approaches to this cell differentiation framework: the first one (already present in the literature) is focused on a network analysis representing the average wandering of the system among its attractors, whereas the second (new) approach takes into consideration the dynamical stories of thousands of individual cells. Results showed that under a particular noise condition the two approaches produce comparable results. Therefore both can be used to model the cell differentiation process in an integrative and complementary manner.

1 Introduction

Cell differentiation is the process by which the development of specialized cell types takes place, starting from a single cell (the zygote). The development of different cell types is the result of highly complex dynamics between intracellular, intercellular, external and inherited signals [4, 5]. Intracellular interactions are captured in gene regulatory networks (GRNs): complex networks that regulate the gene expression. Each cell type presents a particular pattern of gene expression.

Boolean networks (BNs) [6] are models of gene regulatory networks and are prominent examples of complex dynamical systems. Recently a cell differentiation model based on Boolean networks subject to noise has been proposed [11, 12]. This model reproduces the generic abstract features of the differentiation process, such as the attainment of different degrees of differentiation,

deterministic and stochastic differentiation, reversibility, induced pluripotency and cell type change [12]. The model considers the asymptotic behaviours of noisy random Boolean networks, where (intracellular) noise is modelled as the transient flip of a node value. Attractors of BNs are unstable with respect to noise even at low level [10]. In fact, even if the flips last for a single time step sometimes we observe transitions from an attractor to another one. The main abstraction introduced in the model presented in [11, 12] is the Threshold Ergodic Set (TES). TESs represent the asymptotic states of the BNs subject to noise. The various steps of the differentiation process are represented by TES landscapes, which are the emergent results of intracellular noise changes. This model offers a way to mitigate the intrinsic complexity of the analysis of stochastic systems: by applying it we are able to analyse a noisy random Boolean network and produce a static global picture of the all possible differentiation pathways that it can express. So, the main characteristics of the differentiation are captured by TES-based differentiation trees, TES-trees in the following.

The generic abstract properties of the model have been already shown to match those of the real biological phenomenon. However, we remark that *(i)* the results produced by this model depend on the specific noise mechanism implemented and therefore the properties enlighten by the TES model might differ from those observed in the dynamics of real biological cells, as noise acting on them might perturb them in different ways; and *(ii)* the differentiation picture the TES model produces summarises all the possible outcomes of the BN dynamics that may happen under this specific noise mechanism and so it might not represent in sufficient detail individual cell dynamics. For these reasons, we compared the properties of the TES model with the actual dynamical simulation of the BN subject to random external perturbations with the aim of assessing to what extent the two approaches exhibit comparable results and their respective strengths and weaknesses.

The paper is organised as follows. In Sect. 2 we introduce the differentiation model. In Sect. 3 the approach based on stochastic simulations of Boolean networks is illustrated. The experimental setting is described in Sect. 4. Results and discussion are presented in Sects. 5 and 6, respectively.

2 TES Differentiation Model

The cell differentiation model we consider in this work has been presented in [11, 12]. This abstract model¹ is able to describe the most relevant features of the differentiation process, which are the following:

1. *Different degrees of differentiation*: totipotent, pluripotent, multipotent and fully differentiated cells.
2. *Stochastic differentiation*: a population of identical cells can generate different cell types, in a stochastic way.

¹ It is abstract because does not refer to a specific organism or cell type.

3. *Deterministic differentiation*: activation or deactivation of specific genes or group of genes can trigger the development of a multipotent cell into a well-defined type.
4. *Limited reversibility*: a cell can come back to a previous stage under the action of appropriate signals.
5. *Induced pluripotency*: fully differentiated cells can come back to a pluripotent state by modifying the expression level of some genes.
6. *Induced change of cell type*: the expression of few transcription factors can convert one cell type into another.

This differentiation model is based on noisy random Boolean networks. A Boolean network (BN) is a genetic regulatory network (GRN) model, and a complex dynamical system, introduced by Kauffman [6]. A BN is a discrete-state and discrete-time dynamical system whose structure is defined by a directed graph in which each node represents a gene; genes are binary devices that have incoming arcs from other nodes if these last influence the activation of that gene. The most studied BN models are characterized by synchronous dynamics and deterministic functions. With such dynamics, the reachable asymptotic states are *fixed points* and *cyclic attractors*.

This differentiation model takes into account only intracellular noise, since it deals with a single cell as a closed system. It is generic and in principle can support different definitions of noise; however in this contribution we adopt the noise type originally presented in [11, 12]. Hence, we investigate the asymptotic dynamics of BNs subject to noise modelled by the transient flip of a randomly chosen node which lasts for a single time step (a logic negation of node's state). After the transient flip the BN evolves according to its usual deterministic rules until an attractor is found. This noise type represents the smallest stochastic perturbation that can affect a Boolean network; even in this configuration we can observe jumps from an attractor to another one. By perturbing each node of each phase of each attractor found (one at a time), and checking in which attractor the dynamics lead we can compute the *attractor transition matrix* (ATM). This procedure is described in [8, 12, 13]. The ATM summarises the observed transitions between attractors and gives us an estimate of the probabilities with which such transitions can occur; a measure of the system's robustness respect to a random flip of an arbitrary state.

The **Threshold Ergodic Set (TES)** is the key concept introduced on ATM: indeed, cell types are modelled by TESs. A TES_θ is a set of attractors in which the dynamics of the network remains trapped, under the hypothesis that attractor transitions with probability less than threshold θ are not feasible². TESs are computed from the ATM, by iteratively removing the entries with value less than a threshold θ , which is progressively increased from 0 to 1. The TES-trees are constructed following this procedure: TES_0 represents the level 0 and each

² This hypothesis is supported by the observation that cells have a finite lifetime, which enables their dynamics to explore only a portion of the possible attractor transitions.

subsequent level is created if the current threshold applied to the ATM produces a different TES-landscape with respect to the previous one. In this way we capture, in a static representation, all the possible differentiation dynamics of a BN subject to noise. The **threshold** abstraction plays an important role, as it is a mathematical concept strictly related with the noise level in the cell: it scales with the reciprocal of the noise level. High levels of noise (low threshold values) correspond to pluripotent cell states, where the BN trajectory can wander freely among the attractors; conversely, low levels of noise (high thresholds) induce low probabilities to jump between attractors, thus representing the case of specialised cells [11, 12].

3 Stochastic Simulation Approach

The main contribution introduced by the previous model is that the differentiation process is strongly correlated with the *intracellular* noise level. From the model point of view we know how the threshold is related to noise, see [11], and in addition we know that pluripotent cells have a more intrinsic noise level than the more specialized ones [9]. But the threshold and above all its variation mechanism introduced in the model (with which we model the differentiation process) are *externally controlled*. In fact the threshold represents an abstraction of the mechanisms implemented by the real cell to control noise. The identification of autogenous mechanisms, somehow bound to cell's dynamics, through which achieve a threshold self-regulation is subject of ongoing work. As first step to identify the biological mechanisms that affect noise level, and in turn the threshold, we can take in exam a system with different types of noise and noise levels and we can verify if the system is able to reproduce the TES phenomenology. In fact, the approach to cell differentiation previously presented might not capture the real asymptotic configurations of real cells if the cellular system is subject to a noise implemented in a different way with respect to the original model. For example, a real cell dynamics might quickly diverge from the TES model's prevision if its dynamics is such that:

- more than one noise events can occur simultaneously in an asymptotic state;
- noise events occur in its transients.

In addition, the TES-based differentiation trees are constructed following a specific process of threshold variation on the ATM. This process allows us to observe all the differentiation pathways the GRN model is capable of expressing, under a particular noise setting.

To verify to what extent can the TES model predict the entire spectrum of scenarios produced by the dynamics of a system subject to intrinsic noise, we perform time evolutions of Boolean networks subject to different noise levels and we compare these two approaches. Noise levels are represented by distinct frequencies of random perturbations. In such a way, we have the means for counting—for each noise level—the number of differences between the outcomes obtained with the TES model and the stochastic simulations. In the following we call a *story*

a single time evolution of a BN subject to random perturbations. Considering that we are interested in the asymptotic behaviour of the BN dynamics we count the jumps between attractors obtained in each story and we compare them with each level of the TES-based differentiation tree, computed using the TES-model approach on the same BN. We call an *incompatibility* a jump between attractors that would not be allowed given the TES-landscape of a tree’s level.

4 Experimental Setting

The Boolean networks used in the experiments have $n = 100$ nodes and $k = 2$ distinct inputs per node assigned randomly (self-loops are not allowed). Boolean functions have been set by assigning a 1 in the node truth table so as to attain exactly a frequency of 0.5 across all the truth tables (for $k = 2$, this corresponds to the critical value [1]). The rationale behind this choice is that in preliminary results, by setting the bias for each boolean function, in some instances the average overall bias calculated on all nodes could have a non-negligible standard deviation from the desired mean value. Because we want to estimate the differences between the model and the stochastic simulations, we did not want the results to be affected by variance in network dynamic regime. So we use an exact bias, following this procedure for generating networks: we generate a vector of length equal to the sum of the number of Boolean functions’ entries of all nodes in the network ($2^k * n$), we assign half the values to 1 and half to 0 and we use a random permutation of this vector to define the Boolean functions.

The BN is subject to a synchronous dynamics, i.e. all nodes update their state in parallel and functions are applied deterministically. Given that the typical time needed to transcribe a gene is equal to 25–50s in yeasts and 2–3 min in mammals (see reference BNID 111611 [7]); we assume 1 min as a plausible mean value for a BN’s synchronous step of update. In addition, analysing the cell’s average life span in humans (see reference BNID 101940 [7]) we set to 5×10^4 the number of steps for a BN run, in order to model an upper bound of plausible mean cell lifetimes (approximately one month). The only stochastic component resides in the noise, which has been simulated as a temporary flip of the value of a node applied with probability ν ; hence, at each step of the temporal evolution of the network, νn nodes are flipped **on average**. We ran experiments with ν so as to have on average one flip every τ steps, with $\tau \in \{1, 5, 10, 15, 20, 50, 100, 200, 500, 10^3, 5 \times 10^3, 10^4, 2 \times 10^4, 5 \times 10^4\}$. In the following, we will denote the corresponding noise probabilities as ν_τ . Note that the higher τ , the lower the probability ν applied to each node. This noise mechanism emulates possible temporary fluctuations in the expression level of genes and may occur both during stationary phases (i.e. along attractors of the BN) and transients. We run experiments with 30 random BNs; for each of them we compute the ATM and then the TES-tree, following the procedures mentioned in Sect. 2. A typical TES-tree is depicted in Fig. 1. The time evolution of each BN was also simulated 100 times (100 *stories*), each one of them starting from a random initial state. We collected the trajectories of the BNs and computed

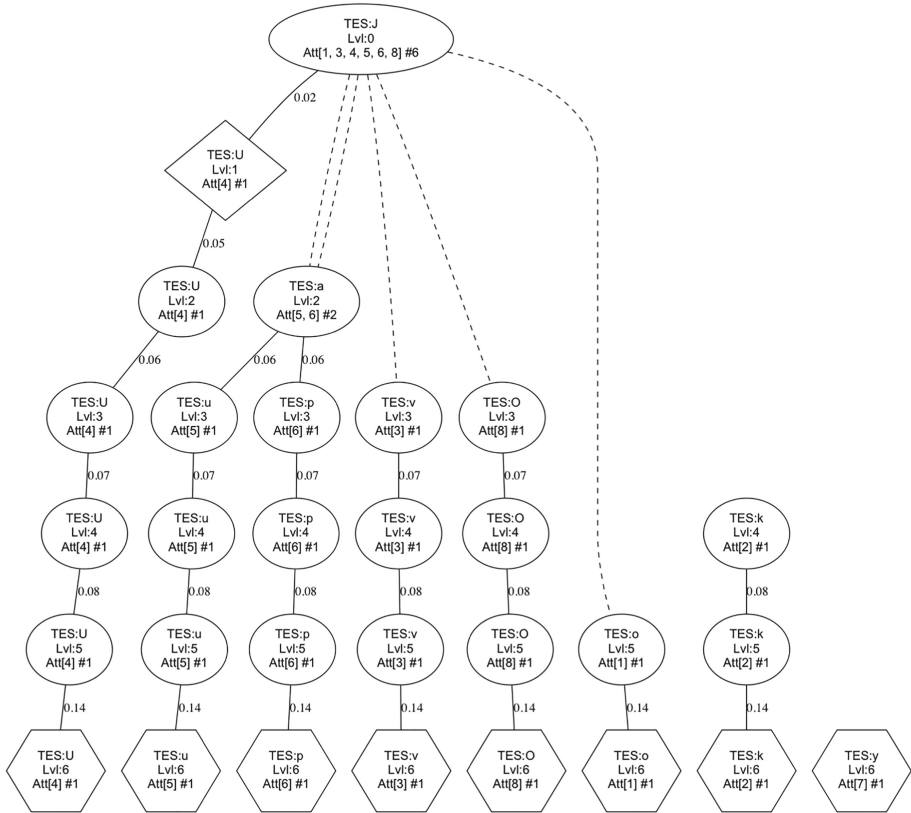


Fig. 1. An example of a TES-tree. Levels are numbered from 0, the topmost, to n , the lowermost; $n = 6$ in this example. TES of level 1 has a diamond shape whereas TESs of level n have a hexagonal one. Labels on the edges indicate the minimum threshold value at which any TESs of the previous level splits or reduces. Continuous lines denote paths along the differentiation tree that can be followed by increasing the threshold at minimum steps (these values are directly obtained by the ATM). Dashed lines denote the paths that can instead be followed if the threshold was increased by larger steps.

statistics on the compatibility between the stories and the TES-tree, besides other ancillary statistics on the overall dynamics of the BNs.

5 Results

In this section we provide the results obtained. The comparison between TES-trees and simulations with stochastic noise is mainly based on counting the transitions between attractors that are observed in the stochastic simulation but that are not allowed by the ATM, given a probability threshold θ . That is, the analysis of what we have called *incompatibilities* between the two approaches

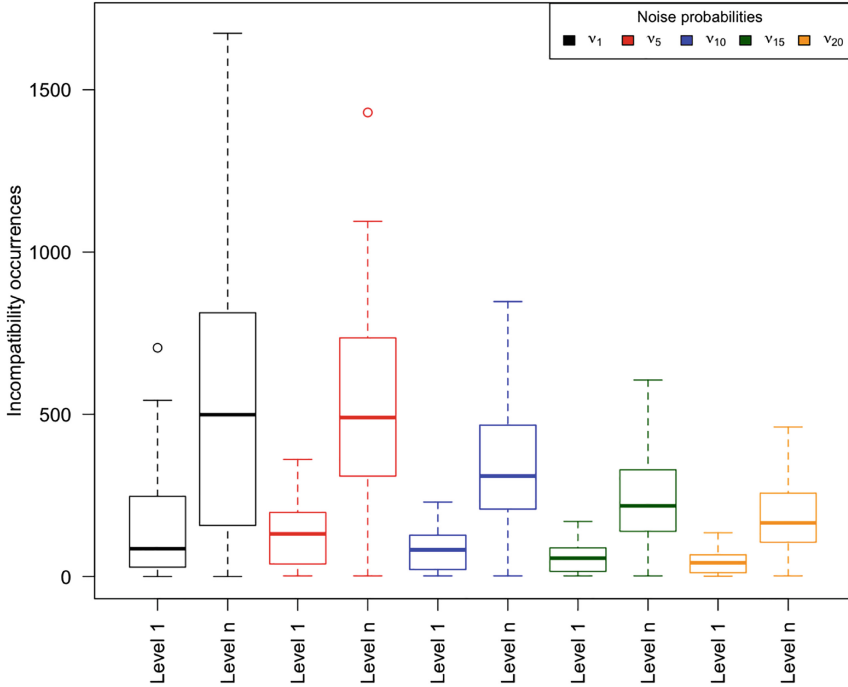


Fig. 2. Distribution of the median values of the *incompatibilities* between the level 1 and level n of the TES-trees and stochastic simulations with the probabilities to flip a node ν so as to have on average one flip every 1, 5, 10, 15, 20 steps. Noise probabilities ν_τ expressed with different colours (Color figure online).

for modelling cell differentiation. For each value of ν_τ , we counted the incompatibilities observed in all the 100 stories w.r.t. the lowest non-zero value of θ (level 1 of the TES-tree) and the highest one, where all TESs are single attractors (level n of the TES-tree). These two particular levels are taken as representative elements able to summarise the trend of incompatibilities since level 1 represents the first TES with not trivial constraints and level n is the most constrained one. Results are summarised in Figs. 2, 3 and 4. In these figures the boxplots graphically represent distributions of the median values of the overall incompatibilities (computed on all 30 BNs) with respect to a particular noise level; different noise levels are represented by distinct colours. For each noise level two boxplots are plotted, one for the incompatibilities with respect to the level 1 and one for the level n .

As expected, the higher ν_τ (corresponding to low values of τ), the higher the number of these incompatibilities. Moreover, this increases with θ ; which corresponds to the increase of the TES-tree's depth. Despite the discrepancy which is apparent at high noise levels, we observe that already for medium noise

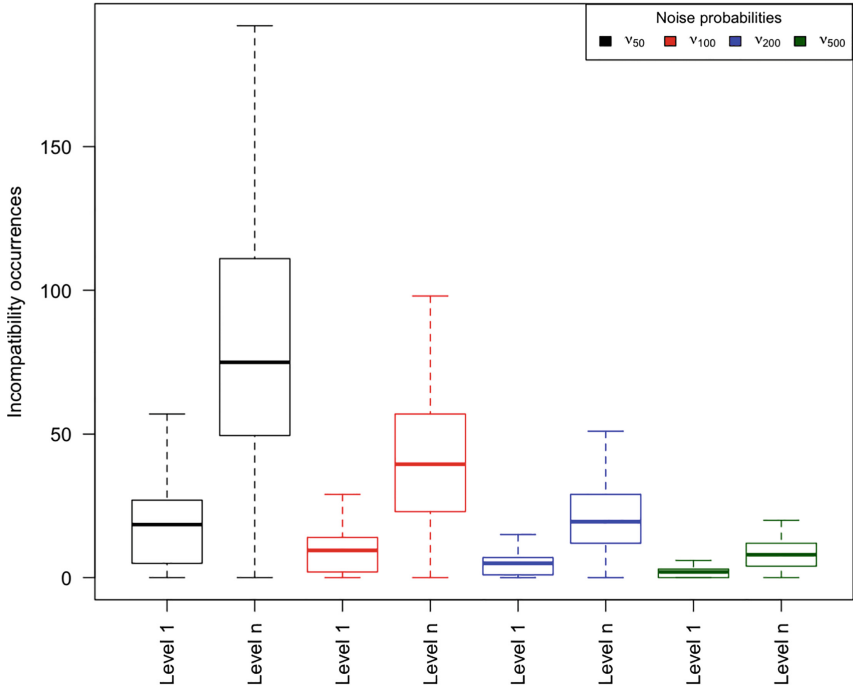


Fig. 3. Distribution of the median values of the *incompatibilities* between the level 1 and level n of the TES-trees and stochastic simulations with the probabilities to flip a node ν so as to have on average one flip every 50, 100, 200, 500 steps. Noise probabilities ν_τ expressed with different colours (Color figure online).

levels, i.e. not higher than ν_{200} , the incompatibilities are limited and tend to be negligible towards low noise levels.

As previously stated, we could observe marked differences between model and simulations if the actual noise presents in the stories is different from that hypothesized by the model. Hence, we analyze the dynamics of the stochastic simulations and we count the number of noise events occurred during transients and the multiple flips in attractors. With multiple flips we mean the occurrence of more than one node value change at a time. Situations both not covered in the model and which could represent the main causes of divergence between the two approaches. In Figs. 5, 6, 7 and 8 each distribution summarises the median values of the property in exam; the median value for each BN computed across the 100 stories of a particular noise level. Hence, we have one boxplot for each distribution of medians. These statistics show that noise events in transients and multiple flips decrease in an exponential way as noise decreases. This trend is more evident in Figs. 6 and 8, which have logarithmic scales. We can note that under noise level ν_{100} the number of multiple flips and noise during transients become negligible with respect to the number of steps considered in the stories

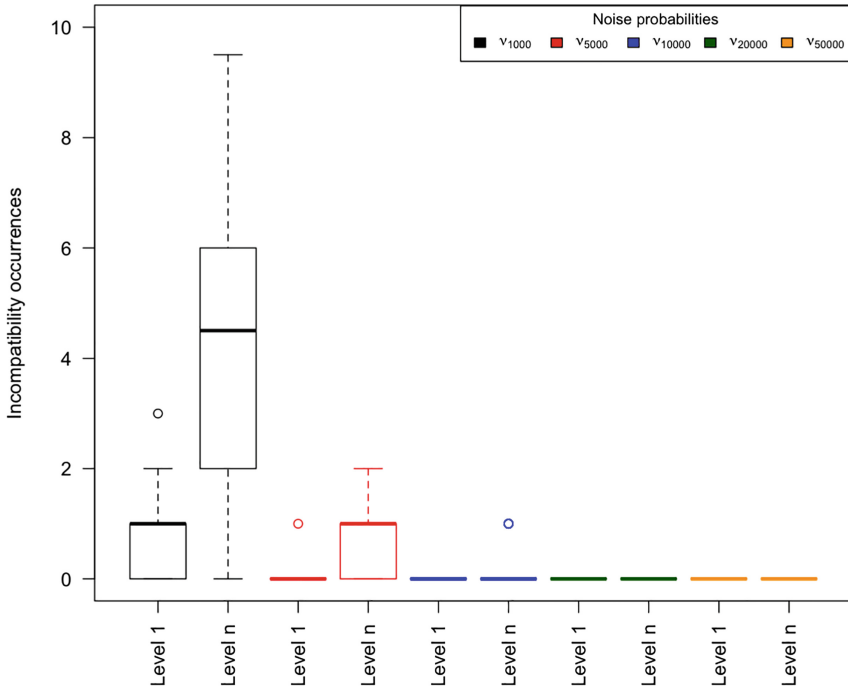


Fig. 4. Distribution of the median values of the *incompatibilities* between the level 1 and level n of the TES-trees and stochastic simulations with the probabilities to flip a node ν so as to have on average one flip every 1000, 5000, 10000, 20000, 50000 steps. Noise probabilities ν_τ expressed with different colours (Color figure online).

(i.e. 5×10^4). We must remark that although the flip of a gene is the smallest stochastic perturbation that can affect a Boolean network it biologically reproduces a fairly intense event, much stronger than molecular fluctuations. Hence, the noise level ν_{200} (250 noise events on average in a story) identified as the convergence point between the two approaches could even be a too high noise level for a real cell's life span. This observation contextualizes the results obtained in a biological framework and it highlights the relevant noise levels in which a real cell can operate.

The results obtained support the statement that there exists a significant noise level under which the two models are in agreement. Therefore, (i) under this threshold they can be both used to model differentiation phenomena—and their observations can be combined—and (ii) the new dynamic simulations may add interesting pieces of information on the heterogeneities of the possible individual configurations.

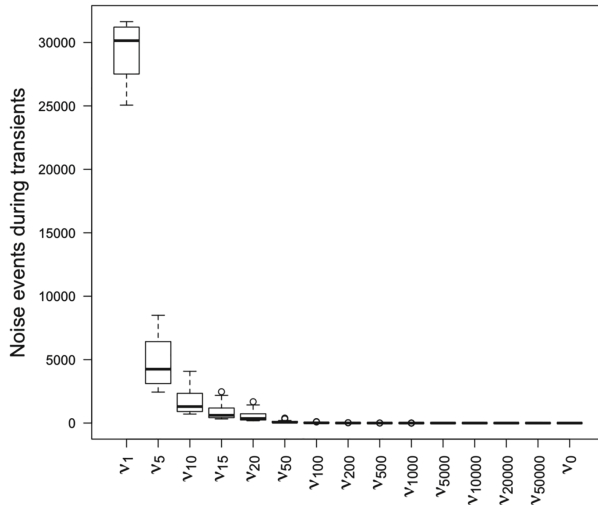


Fig. 5. Distribution of the median values of the number of noise events occurred during transients in stochastic simulations (*stories*), for different noise levels. Noise levels expressed by the ν_τ values in the x axis.

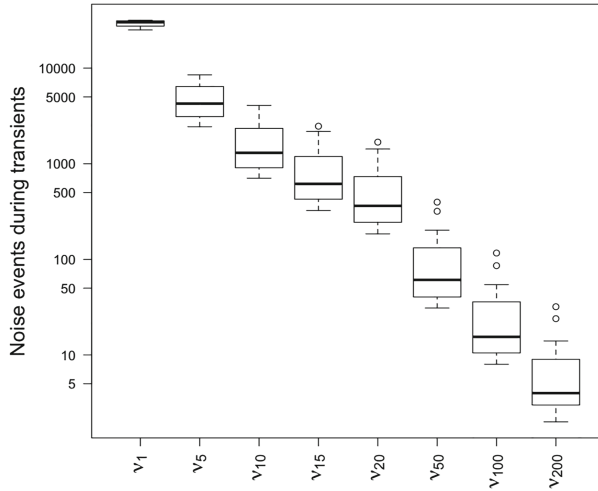


Fig. 6. Detail of Fig. 5 on logarithmic scale.

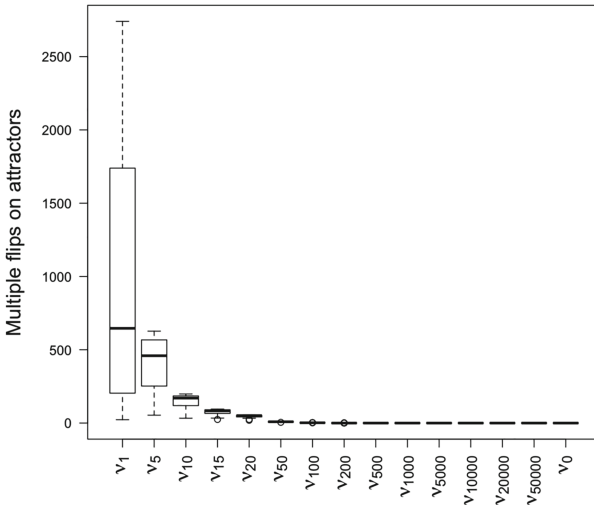


Fig. 7. Distribution of the median values of the number of multiple flips occurred in the attractors in stochastic simulations (*stories*), for different noise levels. Noise levels expressed by the ν_τ values in the x axis.

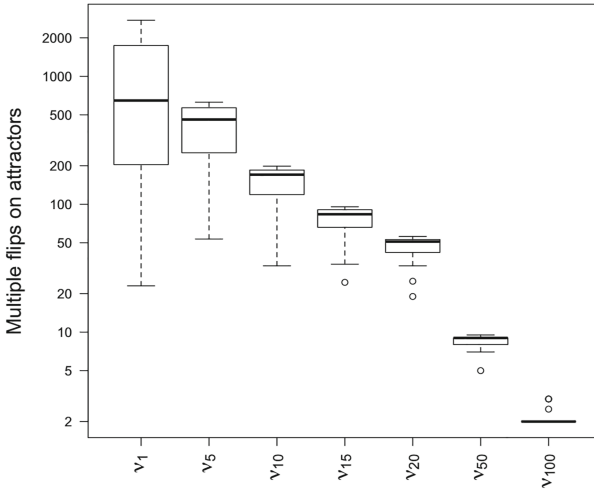


Fig. 8. Detail of Fig. 7 on logarithmic scale.

6 Conclusion

In this paper we have compared two approaches for modelling cell differentiation, both based on random Boolean networks subject to noise. One approach is represented by the well-known model based on TES concept, the other is grounded in time evolutions of BNs subject to different noise levels. The analysis of the emerging differences between these two approaches suggests that there is a specific noise level under which the two models produce similar results. This result has important implications because it shows that both approaches can be used to model cell differentiation and in addition their outcomes can be, at least in part, complementary. Indeed, the new approach could be used to determine the distribution of the extra-cellular noise, due to the intra-cellular events. Moreover this work produced, on the one hand, another proof of robustness of the TES-based differentiation model and, on the other, since the stochastic simulations of BN require less computational cost than the TES model they can be used as an alternative and exploitable approach to conceive more performing automatic procedure for generating biologically plausible cell differentiation model based on BNs [2,3].

References

1. Bastolla, U., Parisi, G.: A numerical study of the critical line of Kauffman networks. *J. Theor. Biol.* **187**(1), 117–133 (1997)
2. Benedettini, S., Roli, A., Serra, R., Villani, M.: Automatic design of Boolean networks for modelling cell differentiation. In: Cagnoni, S., Mirolli, M., Villani, M. (eds.) *Evolution, Complexity and Artificial Life*, vol. 708, pp. 77–89. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-37577-4_5
3. Braccini, M., Roli, A., Villani, M., Serra, R.: Automatic design of Boolean networks for cell differentiation. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016. CCIS*, vol. 708, pp. 91–102. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_8
4. Holland, M.L.: Epigenetic regulation of the protein translation machinery. *EBioMedicine* **17**, 3–4 (2017)
5. Huang, S.: The molecular and mathematical basis of Waddington’s epigenetic landscape: a framework for post-darwinian biology? *Bioessays* **34**(2), 149–157 (2012)
6. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.* **22**(3), 437–467 (1969)
7. Milo, R., Jorgensen, P., Moran, U., Weber, G., Springer, M.: Bionumbers—the database of key numbers in molecular and cell biology. *Nucleic Acids Res.* **38**(Suppl. 1), D750–D753 (2009)
8. Paroni, A., Graudenzi, A., Caravagna, G., Damiani, C., Mauri, G., Antoniotti, M.: CABerNET: a cytoscape app for augmented Boolean models of gene regulatory NETworks. *BMC Bioinform.* **17**, 64–75 (2016)
9. Peláez, N., Gavalda-Miralles, A., Wang, B., Navarro, H.T., Gudjonson, H., Rebay, I., Dinner, A.R., Katsaggelos, A.K., Amaral, L.A., Carthew, R.W.: Dynamics and heterogeneity of a fate determinant during transition towards cell differentiation. *Elife* **4**, e08924 (2015)

10. Ribeiro, A.S., Kauffman, S.A.: Noisy attractors and ergodic sets in models of gene regulatory networks. *J. Theor. Biol.* **247**(4), 743–755 (2007)
11. Serra, R., Villani, M., Barbieri, A., Kauffman, S., Colacci, A.: On the dynamics of random boolean networks subject to noise: attractors, ergodic sets and cell types. *J. Theor. Biol.* **265**(2), 185–193 (2010)
12. Villani, M., Barbieri, A., Serra, R.: A dynamical model of genetic networks for cell differentiation. *PloS ONE* **6**(3), e17703 (2011)
13. Villani, M., Serra, R.: On the dynamical properties of a model of cell differentiation. *J. Bioinform. Syst. Biol.* **2013**(1), 4 (2013)



A Relevance Index Method to Infer Global Properties of Biological Networks

Marco Villani²(✉), Laura Sani¹, Michele Amoretti¹, Emilio Vicari⁴,
Riccardo Pecori^{1,3}, Monica Mordonini¹, Stefano Cagnoni¹, and Roberto Serra²

¹ Dip. di Ingegneria e Architettura, Università di Parma, Parma, Italy

² Dip. Scienze Fisiche, Informatiche e Matematiche,
Università di Modena e Reggio Emilia, Modena, Italy

`marco.villani@unimore.it`

³ SMARTEST Research Centre, Università eCAMPUS, Novedrate, CO, Italy

⁴ Camlin Italy, Parma, Italy

Abstract. Many complex systems, both natural and artificial, may be represented by networks of interacting nodes. Nevertheless, it is often difficult to find meaningful correspondences between the dynamics expressed by these systems and the topological description of their networks. In contrast, many of these systems may be well described in terms of coordinated behavior of their dynamically relevant parts. In this paper we use the recently proposed Relevance Index approach, based on information-theoretic measures. Starting from the observation of the dynamical states of any system, the Relevance Index is able to provide information about its organization. Moreover, we show how the application of the proposed approach leads to novel and effective interpretations in the T helper network case study.

Keywords: Complex systems · Biological networks
Dynamical behavior · Relevance index · T helper cells

1 Introduction

Nowadays a plethora of molecular data results in a vast amount of pathways, networks of interactions and molecular scenarios. A large quantity of information is available on many biological systems, and researchers use it to infer global properties of biological networks [15, 21]. In spite of the strong representational power and flexibility of networks, there are, however, two major limitations which affect most studies in the field [16, 23]:

- the information about the underlying true interactions is often incomplete, so the inferred networks do not provide a complete picture of the interactions in the system under study;
- network studies are often concerned with “static” topological information, like connectivity and betweenness, whereas, in order to understand the functionality of a system, it is important to study its *dynamical properties*.

Modeling the dynamic behavior of such systems is difficult, due to the lack of kinetic data and to computational limitations. Among the methods for facing this problem, those based on steady-state approximations are widely used [13,25]. Nevertheless, these kinds of analysis do not provide enough constraints to find a unique solution to the problem: thus researchers support these techniques by means of suitable hypotheses as, for example, minimization or maximization issues [25]. This drawback, in terms of modeling, has turned out to be particularly relevant when controlling the steady-state behavior of complex networked dynamical systems. In this respect, some efficient model-free methods based on multi-agent reinforcement learning [5] and on mean-field game theory [3] are rapidly emerging in several domains, such as telecommunications.

In order to overcome the aforementioned limitations of steady-state methods, it is worthwhile to resort to methods able to directly deal with the dynamical repertoire of the system. In this paper, we use a recently proposed approach, the *Relevance Index* (RI for short) method [11,32,33], which has the following features:

1. It is based on the observation of the dynamical states of the system (whether simulated or real), without requiring any *a priori* knowledge of the interactions among variables (whenever such knowledge is available, it can be used to complement the proposed method);
2. It can be applied to states coming from different steady state conditions, or even to states obtained from perturbation of these conditions (it does not require fixed asymptotic states);
3. It provides information about the organization of the system itself; indeed, complex systems often display complex organizational features that cannot be captured by a simple tree-like structure;
4. It is robust against noisy or incomplete data, being based on information-theoretic measures.

The overall contribution of this paper is twofold.

On one hand, we show that (i) the dynamically relevant groups of variables identified using the RI index in a biological network are extremely useful in describing the overall dynamics of the system and that (ii) this description could significantly enlarge the explicative power of the graph description of a biological system, by highlighting the links that are really effective.

On the other hand, we present a novel method for creating the homogeneous system used as a reference to evaluate the significance of the RI results. This method considers non-zero pairwise correlations among the variables of the system and is based on the NORTA technique.

The rest of the paper is structured as follows. Section 2 presents the context about complex systems and related works. Section 3 provides a brief review of the Relevance Index method and of the improvement in computing the homogeneous system. Section 4 shows how the application of the RI method leads to novel and effective interpretations in a biological network (T helper case study). Finally, Sect. 5 seals up the work.

2 Context and Related Work

In most natural or artificial dynamical systems, there are groups of variables showing highly coordinated internal dynamics able to significantly influence other groups or even the whole system (Relevant Sets, or shortly RS in the following). The capacity of detecting their presence can often lead to a high-level description of the dynamical organization of the system, and thus to its understanding [32].

However, the identification and monitoring of the significant or relevant portions of dynamical systems is very difficult, especially if these systems exhibit emergent or self-organizing phenomena, the latter being the most interesting and prominent situation for complex dynamical systems [7].

Indeed, most theories and models take into account only two-level systems and describe the formation of relatively simple dynamical patterns as, for example, the creation of the well-known Bénard-Marangoni hexagonal convection pattern [12]. In this case the two levels involved are those of the water particles and of the hexagonal convection cells. Indeed, the apparatus where the phenomenon takes place (which is, of course, necessary, since it determines some major features of the phenomenon itself) is not affected by what happens at the lower levels: in other words, it just provides the fixed boundary conditions that allow the phenomenon to occur.

However, the most interesting recurrent patterns of interaction [18] take place very often at levels that can be regarded as intermediate between pre-existing layers, which are, in turn, affected by the dynamics of these patterns. There are several examples of these “sandwiched” phenomena in physics, biology and social sciences [18]. Perhaps the most evident cases are the presence of vortexes on fluids surfaces, the presence of organs and tissues in multi-cellular organisms, or the action of various groups of humans (such as companies, cooperatives, associations, factions, communities) within societies¹. Note that the formation of structures or patterns not explicitly designed is frequent even in artificial systems, as for example power grids [34], e-mail networks [6], Internet [1, 8], and so on. Thus, the detection of intermediate-level structures and patterns is a very central issue in complex dynamical systems.

Many interesting systems can be represented, at least partially, by means of graphs. In this case, a widespread property is the presence of the so-called communities, portions of system elements within which the connections are dense, but between which they are sparser [20]. Their identification sometimes could detect groups that can be good relevant set candidates.

A method that mixes static and dynamic issues was proposed by Thomas et al. [27, 28] for regulatory networks, the focus of this paper, to capture the main qualitative features of the dynamics of such systems.

¹ The lower and upper level being constituted by the fluid particles and their global stream, by cells and the organism to which they belong, and by human beings and societies, respectively.

Works that use dynamical features in order to detect functional groups are not so frequent; many of them rely on similarity measures and clustering algorithms. This is what is done by Feldt et al. [9], for example.

An interesting approach uses methods introduced in information theory and applied in neurosciences by Edelman and Tononi in 1994 and 1998 [29,30] to detect functional groups of brain regions. In our previous works, we extended the approach to non-stationary dynamical regimes, in order to apply the method to a broad range of systems, including abstract models of gene regulatory networks and simulated social [10], chemical [32], and biological [33] systems. The resulting approach could also be used to identify the critical states of complex dynamical systems [24].

Finally, an interesting literature review about the reconstruction of gene regulatory networks and the development of mathematical models of how the patterns of activation and inhibition determine the state of activation of the network can be found in [4]. The T helper regulatory network considered in this paper is based on the one described in [19].

3 Method

The technique employed in this paper to identify subsets of nodes that are good candidates as RSs is mainly based on the Relevance Index (RI) method. For a complete overview of the methodology adopted in this work please refer to Villani et al. [33]. In the following we will only summarize it briefly.

Main assumptions:

- the values of the system nodes, or variables, express the observed states of the system;
- there exist one or more subsets where these variables are acting in a coordinated way;
- the variables of each subset interact with the other system variables more weakly than among one another internally;
- The computation of the RI is usually based on observational data, and probabilities are estimated as the relative frequencies of the values observed for each variable.

Consider a system U composed of n random variables (X_1, X_2, \dots, X_n) , and a subset S_k composed of k of them, with $k < n$. The $RI(S_k)$ value is defined as the ratio between the *integration* I of S_k and the *mutual information* MI between S_k and the rest of the system:

$$RI(S_k) = \frac{I(S_k)}{MI(S_k; U \setminus S_k)} \quad (1)$$

where $I(S_k)$, the integration, measures the statistical independence of the k elements in S_k and $M(S_k; U \setminus S_k)$, the mutual information, expresses the mutual

dependence between the subset S_k and the rest of the system $U \setminus S_k$. The integration is defined by the following formula:

$$I(S_k) = \sum_{s \in S_k} H(s) - H(S_k) \tag{2}$$

Values of MI equal to zero indicate that the Candidate Relevant Set (CRS in the following) does not communicate with the rest of the system, i.e., it is a separate system and its variables can be neglected. The RI scales with the size of the CRS, thus it needs to be normalized by dividing each member of the quotient in Eq. 1 by its average value within a system where no dynamical structures are present, i.e., a *homogeneous system* where no specific interaction within groups of variables can be highlighted. Moreover, the statistical significance of RI differences should be assessed by means of an appropriate test. For these reasons, a statistical significance index T_c was introduced, which measures how much larger (or smaller) the RI of a subset of variables S_k is with respect to the average RI of groups of the same size within the homogeneous system:

$$T_c(S_k) = \frac{RI(S_k) - \langle RI_h \rangle}{\sigma(RI_h)} = \frac{\nu RI - \nu \langle RI_h \rangle}{\nu \sigma(RI_h)} \tag{3}$$

where $\langle RI_h \rangle$ and $\sigma(RI_h)$ are, respectively, the average and the standard deviation of the RI of a sample of subsets of size k extracted from a reference homogeneous system U_h , and $\nu = \langle MI_h \rangle / \langle I_h \rangle$ is its normalization constant. A more detailed description can be found in previous work [26,31].

The generation of the homogeneous system is critical, and often, in past papers, a simple but general and easy to compute solution was chosen. This solution encompassed the computation of the frequency of each variable, given the available observations, and the generation of a new random series of samples, where each variable had a prior probability equal to the frequency of the original observations. The homogeneity required by Tononi was achieved by considering the components of the random vector U_h , representing the homogeneous system, to be independent. This produced:

1. A unity correlation matrix of the homogeneous system, i.e., with pairwise correlations set to zero;
2. An integration $I(S_k) = 0$ for all subsets of the homogeneous system.

In this paper, we introduce, for the first time, a novelty in the generation of the homogeneous system compared to previous works: homogeneity is maintained by forcing all off-diagonal elements of the correlation matrix to have the same constant value ρ different from zero:

$$CORR(U_h) = \begin{bmatrix} 1 & \rho & \dots & \rho \\ \rho & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \rho \\ \rho & \dots & \rho & 1 \end{bmatrix}$$

Such a value ρ is computed as the average of all pairwise correlations of the observed variables. In this way we preserve both homogeneity and dependence among the different variables.

In order to generate a homogeneous system with the aforementioned features, we use the NORTA method [2], a mathematical procedure that solves the issue of creating random vectors of correlated samples, given the set of their marginal distributions (marginals) and a measure of the dependence among them. The dependence measure we used in NORTA is the usual *product-moment* correlation matrix, based on the linear Pearson correlation coefficient.

As a final step of our methodology, a further *sieving algorithm* [11] can be used to isolate the most representative CRSs, i.e., those having the highest T_c . This procedure is based on the following criterion: if CRS C_1 is a proper subset of C_2 and ranks higher than CRS C_2 , then C_1 is considered to be more relevant than C_2 . Thus it is possible to keep only those CRSs not included in or not including any other CRS with higher T_c . The sieving activity stops when no more eliminations are possible and the remaining sets of variables are the true relevant sets.

4 Experimental Results

4.1 The T Helper Cell Differentiation System

The vertebrate immune system is composed of several cell populations, including antigen presenting cells, natural killer cells, and B and T lymphocytes. There are two main kinds of T lymphocytes: the T cytotoxic cells that actively destroy virus-infected cells and tumor cells and the T helper cells (Th) that take part in cell- and antibody-mediated immune responses by secreting various cytokines, differently distributed in the two main T helper cell sub-types Th1 and Th2. Both sub-types derive from a common precursor Th0 through a rather complex differentiation path, modeled in [19, 22]. In this work, we use the discretization of an updated version of these paths described in [19] (Fig. 1).

The nodes TCR, IL_{18} , IFN_b and IL_{12} receive their input from outside the Th differentiation system and constitute the way the system is aware of its context (in other words, they constitute the system “sensors”). Several signalling pathways are stimulated by their activation [14].

4.2 RI Results

We simulated the gene regulatory network described in Fig. 1 by means of a synchronous Boolean system. There are 2^{19} different initial conditions for each of the 2^4 different scenarios identified by the “sensor” nodes. However, we found only 33 different asymptotic behaviors (all fixed points). Three of these attractors coincide with the gene expression of Th0, Th1 and Th2 cells. These attractors are presented in [19] as the only really stable states, according to the information derived from the application of the so-called generalized logical analysis [28] to the Th differentiation system.

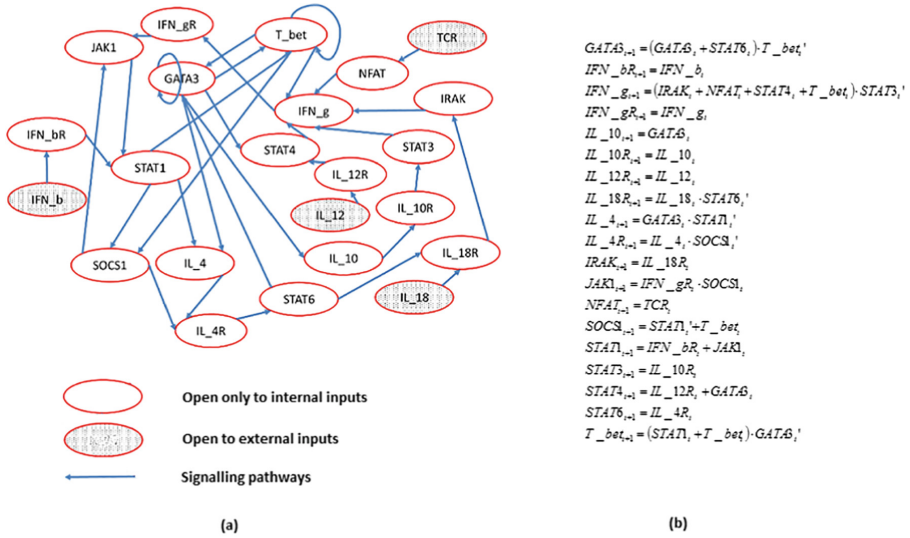


Fig. 1. (a) A graph representation of the Th differentiation system. Note that all the gray-filled nodes (TCR, IL_18, IFN_b, and IL_12) do not receive their input from the network regulating the differentiation system. Thus, in this representation, they do not have incoming links. (b) The dynamical rules of the Th differentiation system as described in [19].

However, the gene regulatory network can express 33 different asymptotic behaviors. Indeed, this fact should give us some information about the dynamical organization of the system². Therefore, to extract this information, we tried to apply the RI methodology (i) to the mere juxtaposition of these attractors or (ii) by weighting their presence proportionally to the size of their basins of attraction, i.e., the width of the neighborhood from which the system converges into the state represented by the attractor under consideration.

In both cases the relevant subsets that were found are composed by TCR and NFAT nodes (Group1 in Fig. 2) and all the other nodes (Group2 in Fig. 2)³.

² In this work we do not make hypotheses about the biological plausibility (or stability or biological function, if any) of these attractors, suggesting the interested readers to refer to Mendoza and Xenarios [19] and to the references quoted therein. Rather we highlight that, once a mathematical model has been established, its structure implies the presence of a well-defined set of attractors: so, an analysis that takes into account their presence (and therefore which highlights their interrelated dynamical relationships) should provide better results than a method that does not act in this way.

³ The node JAK1 is constantly inactive in all attractors. Thus, its presence is useless for the purposes of a dynamical analysis and no CRS include it. Indeed, it is active in transient states, but this kind of analysis is out of the scope of this work (see [24] for a first comparison of the results of RI application to transients and asymptotic states).

Process	Group	TCR	IL_18	IFN_b	IL_12	GATA3	IFN_br	IFN_g	IFN_gR	IL_10	IL_10R	IL_12R	IL_18R	IL_4	IL_4R	IRAK	JAK1	NFAT	SOCS1	STAT1	STAT3	STAT4	STAT6	T_bet	Tel	
Sieve1	Group1																								61379.40	
	Group2																									3519.58
Sieve2	Group3																									1416.54
	Group4																									1186.76
	Group5																									784.15
	Group6																									780.80
Pre-Sieve2	Group7																									642.18
	Group8																									632.36

Fig. 2. The table shows the groups detected by the application of the RI methodology followed by the sieving algorithm (groups 1–6): each group is represented as a row where black boxes denote the variables belonging to it. Group7 and Group8 have been discarded by the sieving algorithm, because they include the stronger relevant subsets indicated as Group3 and Group4: however their observation is important, because it traces a significant coupling among Group3 and Group4 and the other system variables. Indeed, a second application of the iterated RI method fixes this strong association (data not shown).

This fact indicates that the Th differentiation machinery is indeed highly integrated. We can register the presence of these two first CRSs and successively filter them out, in order to apply the sieving algorithm to all the remaining groups. In this case, the two approaches produce different results.

The simple attractor juxtaposition separates Group2 into two big subsets (see Fig. 3, left), whereas the application of RI to an extended set of observations obtained by repeating input data related with the 33 attractors a number of times proportional to the width of their basins of attraction is able to identify (i) the four chains that transmit the external signals toward the inner core of the Th differentiation system (the TCR-NFAT chain, i.e., the Group1, already identified during the first RI application) and (ii) a “circle” of nodes that appears to be the “dynamical engine” of the Th differentiation system, denoted as Group5 (Fig. 3, right).

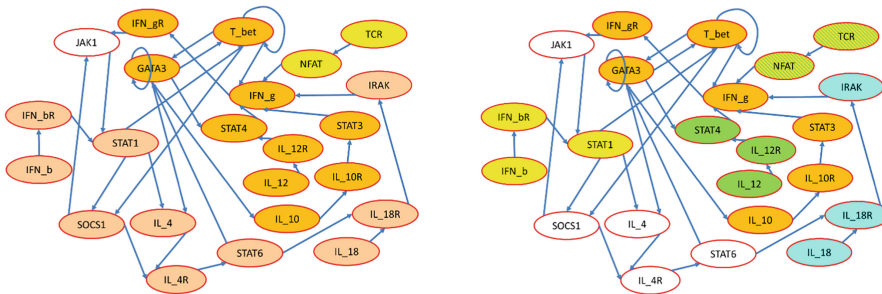


Fig. 3. The main relevant subsets identified using the simple juxtaposition of the attractors of the Th differentiation system (left) or by weighting their presence proportionally to their basins of attraction - the right part of the figure. In this part we highlight the presence of Group1, Group3, Group4, Group5, and Group6, respectively in striped, yellow, blue, orange, and green background. (Color figure online)

It appears that nodes SOCS1, IL_4, IL_4R, and STAT6 do not belong to any relevant subsets (Fig. 3, right), if we strictly adhere to the relevant subset definition. However, before the application of the sieving algorithm, the RI analysis reports two highly-ranked groups in the top positions, namely Group7 (composed by the aforementioned nodes and by Group3) and Group8 (composed of IL_4, IL_4R, STAT6, and by Group4). Indeed, these two groups are discarded by the sieving algorithm because they include two already identified and slightly stronger relevant subsets. Vice-versa, we can use this information in order to identify the nodes influenced by (or influencing) Group3 and Group4. Thus, given the directions of the links of the Th system, it appears that the information acquired by Group3 (in particular by node IFN_b) is transmitted to the nodes belonging to the “white group”, which, in turn, passes it to Group4. Therefore, the white group is composed by elements that seem to act as a sort of “transmission engine” for the Th differentiation system. Figure 4 highlights such an information flow from the “yellow” region (group 7) to the “blue” region (group 8).

The RI analysis therefore induces an interesting interpretation of the dynamical data which, when mapped on the already available topological knowledge, provides an expressive explanation of the system functioning. The same knowledge (the identification of groups of variables and of their relationships) is not derivable from the static analysis alone. The usual algorithms for the search of communities [17, 20] identify only the pair GATA3-T_bet. Moreover, only one of the identifiable 27 circuits is highlighted (Group5, which involves nodes T_bet, GATA3, IL_10, IL_10R, STAT3, IFN_g, IFN_gR, JAK1 and STAT1).⁴

On the other hand, the usual dynamical analyses are mainly focused on the detailed reproduction or prediction of the system’s behaviors [19] and therefore are not suitable for a highly abstracted and “global” vision of the system functioning. The same generalized logical analysis [28] that mixes topological and dynamical issues identifies chains of positive and negative feedbacks, eventually providing clues for the identification of stable attractors, but does not give the overall vision of the RI method, which identifies the genes involved in injecting information into the system (the groups 1, 3, 4 and 6) and the main circuit responsible of the information processing (group5).

Obviously, this method cannot be used to reconstruct the detailed topology of the investigated system (though it could suggest useful groupings). It is worth mentioning, however, that the RI method can be applied directly to the experimental data, if these are available. In this respect, we can note that while the collection of time series is an experimentally difficult and costly task, the RI methodology can be applied merely by comparing different steady states (whose data could derive even from different beings), in such a way taking advantage from more common data sources. In case experimental data are available, the RI method can provide an effective idea of the dynamical organization of the observed system without requiring any knowledge of topology, dynamical rules, or parameters [26, 31, 32].

⁴ Note that the node STAT1 participates in Group 3, one of the “sensors groups” of the Th differentiation system.

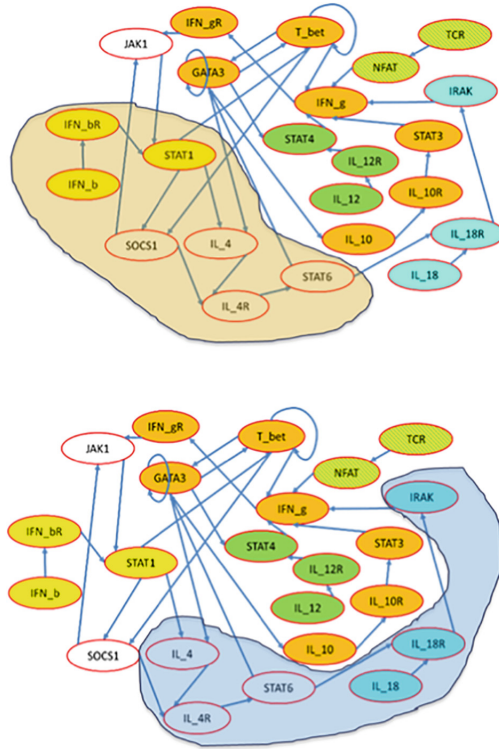


Fig. 4. Same as Fig. 3, but highlighting the correlation of Group3 and Group4 with other Th differentiation variables (SOCS1, IL-4, IL-4R and STAT6 – for brevity indicated in this caption as “WhiteGroup”). With reference to the table reported in Fig. 2, one can see that, indeed, the first (and unique) significant appearance of these variables as a block occurs along with Group3 and Group4, with which they compose Group7 and Group8, as shown by the third block of results in the table. Given the directions of the links, in this example assumed to be known, it appears that the graph structure of the system could allow the signal transmission from Group3 and Group5 to the WhiteGroup (first row). However, the RI index indicates as evident the influence of just Group3. In turn, the information acquired by the WhiteGroup from Group3 is transmitted to Group4, in such a way modulating the external signals coming from node IL-18 (second row). (Color figure online)

5 Conclusion

In this paper, we proposed to use the RI method, improved through a novel technique for computing the correlation matrix of the homogeneous system, as a means to infer global properties of biological networks. With respect to steady-state approximation approaches, the RI method, which is based on the observation of the dynamical states of the system, provides information about the organization of the system itself and is robust against noisy or incomplete

data, being based on information-theoretic measures. The RI method can be applied directly to the experimental data, if available. In this case, it can sketch an effective picture of the dynamical organization of the observed system. As a use case, we illustrated the analysis of the T helper network.

Regarding future work, we plan to apply the RI method to several biological networks. This can be done quite easily because it can be applied to system characterized by both continuous and discrete (Boolean or multi-valued) variables. The ultimate objective is twofold and encompasses both finding new insights about those systems and refining the method itself. In particular, we are interested in studying systems with a large number of nodes, which cannot be explored exhaustively, even with parallel computing approaches. For such systems, the adoption of meta-heuristics is necessary in order to find the relevant groups of nodes in a reasonable amount of time.

Acknowledgments. The work of Michele Amoretti was supported by the University of Parma Research Fund - FIL 2016 - Project “NEXTALGO: Efficient Algorithms for Next-Generation Distributed Systems”.

This work greatly benefited from discussions with Andrea Roli, to whom the authors are warmly thankful.

References

1. Albert, R., Jeong, H., Barabási, A.L.: Internet: diameter of the world-wide web. *Nature* **401**(6749), 130–131 (1999)
2. Cario, M.C., Nelson, B.L.: Modeling and generating random vectors with arbitrary marginal distributions and correlation matrix. Technical report (1997)
3. Cimorelli, F., Priscoli, F.D., Pietrabissa, A., Celsi, L.R., Suraci, V., Zuccaro, L.: A distributed load balancing algorithm for the control plane in software defined networking. In: 2016 24th Mediterranean Conference on Control and Automation (MED), pp. 1033–1040, June 2016
4. De Jong, H.: Modeling and simulation of genetic regulatory systems: a literature review. *J. Comput. Biol.* **9**(1), 67–103 (2002)
5. Delli Priscoli, F., Di Giorgio, A., Lisi, F., Monaco, S., Pietrabissa, A., Celsi, L.R., Suraci, V.: Multi-agent quality of experience control. *Int. J. Control Autom. Syst.* **15**, 892–904 (2017)
6. Ebel, H., Mielsch, L.I., Bornholdt, S.: Scale-free topology of e-mail networks. *Phys. Rev. E* **66**, 035103 (2002). <http://www.citebase.org/cgi-bin/citations?id=oai:arXiv.org:cond-mat/0201476>
7. Emmeche, C., Køppe, S., Stjernfelt, F.: Explaining emergence: towards an ontology of levels. *J. Gen. Philos. Sci.* **28**(1), 83–117 (1997)
8. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On power-law relationships of the Internet topology. *SIGCOMM Comput. Commun. Rev.* **29**(4), 251–262 (1999)
9. Feldt, S., Waddell, J., Hetrick, V., Berke, J., Żochowski, M.: Functional clustering algorithm for the analysis of dynamic network data. *Phys. Rev. E* **79**(5), 056104 (2009)
10. Filisetti, A., Villani, M., Roli, A., Fiorucci, M., Poli, I., Serra, R.: On some properties of information theoretical measures for the study of complex systems. In: Pizzuti, C., Spezzano, G. (eds.) WIVACE 2014. CCIS, vol. 445, pp. 140–150. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12745-3_12

11. Filisetti, A., Villani, M., Roli, A., Fiorucci, M., Serra, R.: Exploring the organisation of complex systems through the dynamical interactions among their relevant subsets. In: Proceedings of the European Conference on Artificial Life, pp. 286–293 (2015)
12. Haken, H.: An introduction. Synergetics, pp. 1–387. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-662-10184-1_1
13. Herrgård, M.J., Covert, M.W., Palsson, B.Ø.: Reconstruction of microbial transcriptional regulatory networks. *Curr. Opin. Biotechnol.* **15**(1), 70–77 (2004)
14. Huang, Y., Wange, R.L.: T cell receptor signaling: beyond complex complexes. *J. Biol. Chem.* **279**(28), 28827–28830 (2004)
15. Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N., Barabási, A.L.: The large-scale organization of metabolic networks. *Nature* **407**(6804), 651–654 (2000)
16. Johnson, J.: *Hypernetworks in the Science of Complex Systems*, vol. 3. World Scientific, Singapore (2013)
17. Johnston, H.: Cliques of a graph-variations on the Bron-Kerbosch algorithm. *Int. J. Parallel Prog.* **5**(3), 209–238 (1976)
18. Lane, D., Pumain, D., van der Leeuw, S.E., West, G.: *Complexity Perspectives in Innovation and Social Change*, vol. 7. Springer Science and Business Media, Berlin (2009). <https://doi.org/10.1007/978-1-4020-9663-1>
19. Mendoza, L., Xenarios, I.: A method for the generation of standardized qualitative dynamical systems of regulatory networks. *Theor. Biol. Med. Model.* **3**(1), 13 (2006)
20. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev. E* **69**(2), 026113 (2004)
21. Ravasz, E., Somera, A.L., Mongru, D.A., Oltvai, Z.N., Barabási, A.L.: Hierarchical organization of modularity in metabolic networks. *Science* **297**(5586), 1551–1555 (2002)
22. Remy, E., Ruet, P., Mendoza, L., Thieffry, D., Chaouiya, C.: From logical regulatory graphs to standard petri nets: dynamical roles and functionality of feedback circuits. In: Priami, C., Ingólfssdóttir, A., Mishra, B., Riis Nielson, H. (eds.) *Transactions on Computational Systems Biology VII*. LNCS, vol. 4230, pp. 56–72. Springer, Heidelberg (2006). https://doi.org/10.1007/11905455_3
23. Serra, R., Villani, M.: *Modelling Protocells*. Springer Science and Business Media, Dordrecht (2017). <https://doi.org/10.1007/978-94-024-1160-7>
24. Roli, A., Villani, M., Caprari, R., Serra, R.: Identifying critical states through the relevance index. *Entropy* **19**(2), 73 (2017)
25. Ruppín, E., Papin, J.A., De Figueiredo, L.F., Schuster, S.: Metabolic reconstruction, constraint-based analysis and game theory to probe genome-scale metabolic networks. *Curr. Opin. Biotechnol.* **21**(4), 502–510 (2010)
26. Sani, L., Amoretti, M., Vicari, E., Mordonini, M., Pecori, R., Roli, A., Villani, M., Cagnoni, S., Serra, R.: Efficient search of relevant structures in complex systems. In: Adorni, G., Cagnoni, S., Gori, M., Maratea, M. (eds.) *AI*IA 2016*. LNCS (LNAI), vol. 10037, pp. 35–48. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49130-1_4
27. Thomas, R., Kaufman, M.: Multistationarity, the basis of cell differentiation and memory. I. structural conditions of multistationarity and other nontrivial behavior. *Chaos Interdisc. J. Nonlinear Sci.* **11**(1), 170–179 (2001)
28. Thomas, R., Thieffry, D., Kaufman, M.: Dynamical behaviour of biological regulatory networks-I. Biological role of feedback loops and practical use of the concept of the loop-characteristic state. *Bull. Math. Biol.* **57**(2), 247–276 (1995)

29. Tononi, G., McIntosh, A.R., Russell, D.P., Edelman, G.M.: Functional clustering: identifying strongly interactive brain regions in neuroimaging data. *Neuroimage* **7**(2), 133–149 (1998)
30. Tononi, G., Sporns, O., Edelman, G.M.: A measure for brain complexity: relating functional segregation and integration in the nervous system. *Proc. Nat. Acad. Sci.* **91**(11), 5033–5037 (1994)
31. Vicari, E., Amoretti, M., Sani, L., Mordonini, M., Pecori, R., Roli, A., Villani, M., Cagnoni, S., Serra, R.: GPU-based parallel search of relevant variable sets in complex systems. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016. CCIS*, vol. 708, pp. 14–25. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_2
32. Villani, M., Filisetti, A., Benedettini, S., Roli, A., Lane, D., Serra, R.: The detection of intermediate-level emergent structures and patterns. In: *Proceedings of the European Conference on Artificial Life*, pp. 372–378 (2013)
33. Villani, M., Roli, A., Filisetti, A., Fiorucci, M., Poli, I., Serra, R.: The search for candidate relevant subsets of variables in complex systems. *Artif. Life* **21**, 412–431 (2015)
34. Watts, D.J., Strogatz, S.H.: Collective dynamics of “small-world” networks. *Nature* **393**(6684), 440–442 (1998)



Dynamical Properties of a Gene-Protein Model

Davide Sapienza¹, Marco Villani^{1,2}, and Roberto Serra^{1,2}✉

¹ Department of Physics, Informatics and Mathematics,
Modena and Reggio Emilia University, Modena, Italy
{marco.villani, rserra}@unimore.it

² European Centre for Living Technology, Ca' Foscari University, Venice, Italy

Abstract. A major limitation of the classical random Boolean network model of gene regulatory networks is its synchronous updating, which implies that all the proteins decay at the same rate. Here a model is discussed, where the network is composed of two different sets of nodes, labelled G and P with reference to “genes” and “proteins”. Each gene corresponds to a protein (the one it codes for), while several proteins can simultaneously affect the expression of a gene. Both kinds of nodes take Boolean values. If we look at the genes only, it is like adding some memory terms, so the new state of the gene subnetwork does no longer depend upon its previous state only.

In general, these terms tend to make the dynamics of the network more ordered than that of the corresponding memoryless network. The analysis is focused here mostly on dynamical critical states. It has been shown elsewhere that the usual way of computing the Derrida parameter, starting from purely random initial conditions, can be misleading in strongly non-ergodic systems. So here the effects of perturbations on both genes’ and proteins’ levels is analysed, using both the canonical Derrida procedure and an “extended” one. The results are discussed. Moreover, the stability of attractors is also analysed, measured by counting the fraction of perturbations where the system eventually falls back onto the initial attractor.

Keywords: Gene-protein model · Generic properties · Memory effect
Dynamical regimes

1 Introduction

Random Boolean models of genetic regulatory networks (RBNs) are very well-known and, in spite of their long age, they still provide useful descriptions of important observational and experimental results [8, 12–17]. A major limitation of the classical RBN model is its synchronous updating: from a physical viewpoint, this amounts at assuming that all the proteins decay at equal rates: this unrealistic assumption allows one to write the gene activation pattern at time $t + 1$ as a function of that pattern at time t , forgetting the previous history. Asynchronous updating has been sometimes proposed (one gene at each time step), but this also leads to difficult interpretations, due to the relatively large typical protein decay time and to the very large number of genes. Other interesting “intermediate” update strategies have also been proposed [5, 19].

Some properties of RBNs are robust with respect to the updating strategy, but in general there is no guarantee that this is the case. In particular, one should be very careful when dealing with the networks' dynamical properties. We have been particularly interested in the response of genetic networks to perturbations like gene knock-out and we have shown that, if the RBN model is chosen, the distribution of avalanches in gene expression levels in *S. Cerevisiae* that follows a single knock-out provides information about the dynamical regime of the biological network [8, 16]. This result is particularly relevant, given the importance of the “criticality hypothesis”, which states that biological systems should preferentially be found in dynamically critical states [13]. If we are indeed interested in biological genetic networks, such issues should be addressed in a way that does not critically depend upon the unrealistic assumption of synchronicity: different updating schemes should be considered, privileging whenever possible those that are closer to what we know about the behaviour of real gene regulatory networks.

In order to do so, while retaining the simplifications related to the use of Boolean variables and to the “generic” approach of RBNs, we introduced the GPBN model (Gene-Protein Boolean Network), where the network is composed of two different sets of nodes, labelled G and P with reference to “genes” and “proteins” [9–11]. It is now well-established that proteins are not the only genetically-encoded products which can influence the effective expression level of other genes (think for example of miRNAs [2, 3]). However, in order to simplify the model description, we will call here “proteins” all the products of gene activation that are able to influence the expression of other genes.

Each gene corresponds to a protein (the one it codes for), while several proteins can simultaneously affect the expression of a gene. Both kinds of nodes take Boolean values: the state at time $t + 1$ of a G node depends upon the state of a fixed set of P nodes at the same time, while the state at time $t + 1$ of a P node depends upon the state of its corresponding G node at time t . Once a P node is set active (its state is 1), it remains active for at least a fixed number of steps. If a new activation signal comes in before decaying, the counter is reset. If no activation signal arrives, the P node is set to 0 at the end of its “lifespan”. If we look at the genes only, it is like adding some memory terms, so the new state of the network is no longer “Markovian”, i.e. it does no longer depend upon the previous state only.

This model has been thoroughly studied and its properties have been described elsewhere [9, 11]. In those papers the usual definition of dynamical criticality, based on the value of the so-called Derrida parameter, had been used. We have recently shown some limitations related to the use of that single measure to characterize critical states in RBNs [4]. In particular, the choice of a completely random initial state in the computation of the Derrida parameter has been criticized and a different measure (“extended Derrida parameter”) has been proposed [18].

This prompted a more thorough analysis of the dynamics of GPBNs, whose main features are presented in this paper.

The paper is organized as follows: in Sect. 2 the GPBN model is described, while in Sect. 3 the measures of dynamical criticality are discussed and the extended Derrida parameter is introduced. In Sect. 4 the results obtained by simulating GPBNs are shown and discussed, paying particular attention to the similarities and differences between the “canonical” (i.e. standard) and the extended Derrida procedures.

A different way to evaluate the robustness of the network behaviour, based upon perturbations of its dynamical attractors, is also presented. Critical discussion and suggestions for further research are summarized in Sect. 5.

2 The GPBN Model

A GPBN model [9–11] is a bipartite oriented graph containing two types of Boolean nodes: the G nodes, which represent the genes set, and the P nodes, which represent the set of proteins (or, in general, gene products). A G node can be active or inactive (producing or not its protein), whereas a P node describes the presence (or absence) of a protein within the system. There are two types of links: *synthesis links*, which go from a G node to only one P node, and *transcriptional regulation links*, from a P node to one or more G nodes.

As usual in RBNs, time evolves in discrete steps. Note that the state at time $t + 1$ of the GPBN model is determined by its state at time t , and the update is formally synchronous. However, due to the presence of the P nodes, the updating of the gene subnetwork is not synchronous, i.e. the states of G nodes at time $t + 1$ are not determined by their states at the previous time step.

Each G node, say the j -th, produces its protein when active (synthesis link) and a G node is driven by the action of its k inputs (k being the number of its transcriptional regulation links, coming from P nodes), according to a fixed Boolean function f_j associated to it ($f_j: \{0, 1\}^k \rightarrow \{0, 1\}$).

The topology of the transcriptional links is random, and so is the choice of the Boolean functions: each f_j is generated by assigning at random to each of its 2^k possible inputs an output equal to 1 with probability p (the so-called bias of the set of Boolean functions), 0 otherwise.

To each P node, say the i -th, an integer non-negative variable h_i is also associated (its *decay phase*) which can change in time and which represents its residual lifetime. The maximum value of h_i is the *decay time* dt_i of node i , representing the lifespan of the protein, once activated (i.e. just synthesized). When a P node is activated, its decay phase h_i takes the value dt_i and it is later decreased by 1 at each time step, until it ends in 0 (unless the same node is not activated again in that time interval). When the incoming G node is active, then the corresponding P node resets its decay phase to the decay time. As long as the decay phase takes a nonzero value, the P node has a regulation role on its outgoing links (i.e. its value in the transition function is 1).

The decay time of each node is taken randomly with uniform probability between 1 and a parameter defined as *maximum decay time* (MDT); note that when MDT is equal to 1 the GPBN is identical to the corresponding RBN (i.e. the one with the same topology and the same activation functions). If the value of a G node is 1 at time t then the value of the corresponding P node will be 1 at time $t + 1$ and its decay phase will be set to dt_i , otherwise the decay phase of the P node is decremented by one unit (in case of $dt_i = 0$, the activation of P is set to 0). On the other hand, the value of the G node at time t is immediately determined by its function f_j , which depends on the states of its incoming P nodes at time t .

3 Dynamical Regimes

The asymptotic states of finite RBNs are periodic cycles; fixed points correspond to cycles with unitary period. Different dynamical regimes have been observed in RBNs [1, 13, 14], classified as disordered (sometimes called “chaotic”, although all the attractors are indeed periodic), ordered or critical depending upon the length of their periods and the sensitive dependence upon initial conditions. In chaotic networks the cycle length sharply increases with the network size, and nearby initial states are likely to lead to different attractors, while in ordered systems the typical cycle length shows a polynomial dependence upon the number of nodes, and basins of attraction are quite regular. Given the random nature of these systems, the analysis usually concerns families of networks built by keeping fixed some parameters, like e.g. the number of nodes, the average number of connections per node and/or the average bias of the Boolean functions, while changing in different network realizations the topology of connections and the transition functions. Critical networks are those whose parameters lie on (or close to) the manifolds that separate regions in parameter space with ordered behaviours from the chaotic regions. It is important to stress that these terms refer to the typical features of networks with those parameters, while a single network realization can behave in a way very different from the typical ones. Large deviations from typical behaviours can easily be found in critical networks [15].

The asymptotic dynamics can be identified by means of the so-called dynamical Derrida parameter λ [6, 7], which measures the tendency of a temporary perturbation to vanish, to persist or to spread through the entire system: so, ordered, critical and chaotic dynamical regimes correspond respectively to $\lambda < 1$, $\lambda \approx 1$ and $\lambda > 1$.

This parameter can be determined by analysing a plot of the average distance between two states at time $t + 1$ versus their distance at time t (the Derrida plot) and by looking at the slope of the tangent to the curve in the limit of small initial distances.

Different (static) measures of the dynamical properties have also been proposed, based on an analysis of the properties of the set of Boolean functions rather than on actual simulations: they are discussed in depth in [18] alongside with their relationships with the dynamical Derrida parameter, described above, which is the only such measure considered in this paper.

Another important remark raised in [18] concerns the dependency of the dynamical Derrida parameter from the set of initial conditions. The usual recipe is that of choosing a fully random initial state, and of considering the time behaviour of its perturbed states. While this is entirely reasonable in ergodic systems (where all accessible states are equiprobable over a long period of time), RBNs with a small number of connections per node are strongly non-ergodic [20], so it may easily happen that such purely random states are never encountered in the life of the cell modelled by the Boolean genetic network.

It seems therefore physically much more appropriate to determine the dynamical Derrida parameter while limiting the set of allowed initial states only to those states that are the successors of some other states. The initial state might be found by starting the network simulation from a purely random state, letting it evolve for T_{ev} steps ($T_{ev} \geq 1$) and by choosing the state that has been reached as the initial state for computing the

Derrida parameter. When the set of allowed initial states is limited in this way, we refer to an “extended Derrida approach”, or to an “extended Derrida parameter”, to distinguish it from the canonical one.

Note also that different types of perturbations are possible: in GPBNs the initial perturbation could affect G nodes, P nodes, or both. In our approach a perturbation of a P node can correspond either (i) to an activity change from 0 to 1, with a decay phase h_i randomly chosen within the range $[1, dt_i]$ or (ii) to an activity change from 1 to 0, with $h_i = 0$. A perturbation of a G node can correspond (i) to an activity change from 0 to 1, followed by the appropriate effect on the protein or (ii) to an activity change from 1 to 0 – in this case, the G node is not producing its protein, and the P node reduces its decay phase by one.

4 Results

It had already been observed in [9, 11] that, as it might be a priori expected, the presence of a memory term tends to make the dynamical behaviour “more ordered”. This can be shown by comparing the behaviour of networks with $MDT \neq 1$ with those of the corresponding network with $MDT = 1$ (that are identical to the corresponding RBNs). The comparison can be made for different dynamical behaviours, in this paper we will report results concerning networks that are critical if $MDT = 1$. Three sets of parameters, all corresponding to critical behaviours, will be discussed: $[k = 2, p = 0.5]$, $[k = 3, p = 0.21]$, $[k = 3, p = 0.79]$. The fact that two different cases are chosen for $k = 3$ is due to the fact that in GPBN the 0–1 symmetry of RBNs no longer holds.

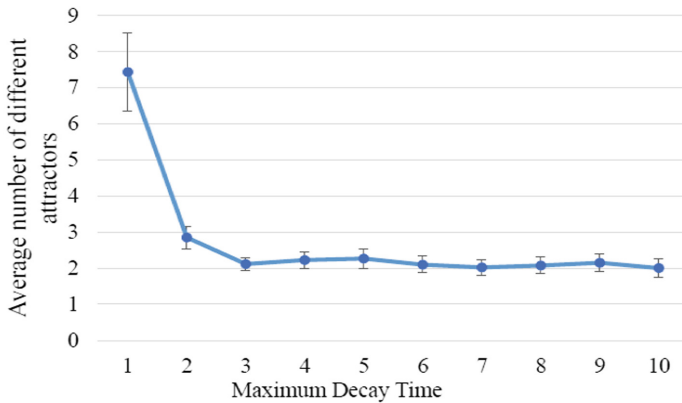


Fig. 1. Number of different attractors vs. maximum decay time (MDT, ranging from 1 to 10); each point represents the average of 1000 different networks (case $[k = 2, p = 0.5]$) with 100 G-P node pairs. For each network 100 runs with different initial conditions are performed, until an attractor (with period lower than 1000 time steps) is reached or until the sum of the transient time exceed 10000 time steps

The stabilizing effect of memory can be seen in Fig. 1, where the number of different attractors versus the maximum decay time is shown to decrease sharply even with a short memory term [9].

Let us now turn to the dynamical regime, as determined by the Derrida procedure. As discussed in Sect. 3, perturbations can be performed either on G or on P nodes. Let us first consider this latter case. In all the simulations described here below the perturbations can be either up (i.e. setting equal to one the value of a P node which is 0) or down, depending on the not perturbed activity of the chosen P node. In each simulation series we create 50 different networks with 100 G-P node pairs, 100 different initial conditions for each network. In order to allow an easier series comparison we consider the decay time of each P node being exactly equal to MDT.¹

In Fig. 2 the behaviour of the Derrida parameter for the critical case $k = 2$, $p = 0.5$ is shown. The two curves refer to the G-node and to the P-node subnetworks. Very large values of MDT have also been considered, and it is shown that the network remains critical notwithstanding the memory term.

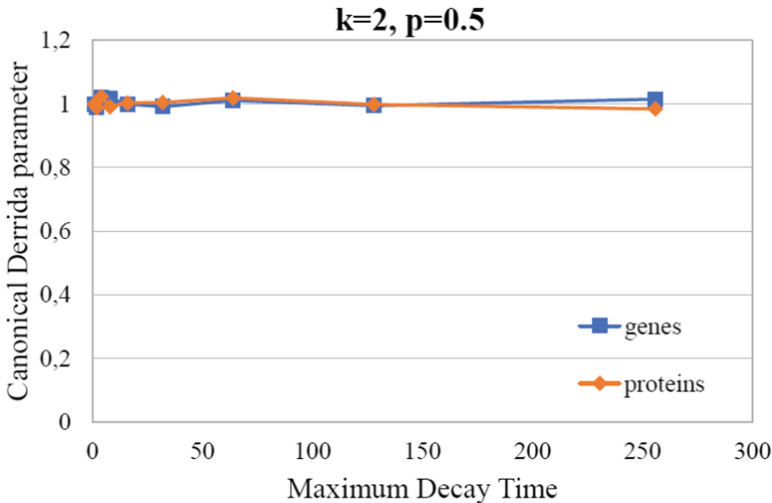


Fig. 2. Canonical Derrida parameter vs MDT (MDT $\in \{1, 2, 4, 8, 16, 32, 64, 126, 256\}$), case $k = 2$, $p = 0.5$. The two curves refer to the G-node and to the P-node subnetworks, subject to a P-node perturbation

In Fig. 3 the same parameter is shown for the two cases with $k = 3$. While the G-node subnetwork remains critical, here the effect of the memory term on the P subnetwork is neither that of leaving it critical, nor that of always bringing it in the

¹ Subsequent simulation series where the decay time of each node is randomly chosen (with uniform probability) in $[1, \text{MDT}]$ show that the main effect of choosing the decay times randomly with uniform probability between 1 and MDT is that of slightly soften the shape of the curves, without altering their behavior (data not shown).

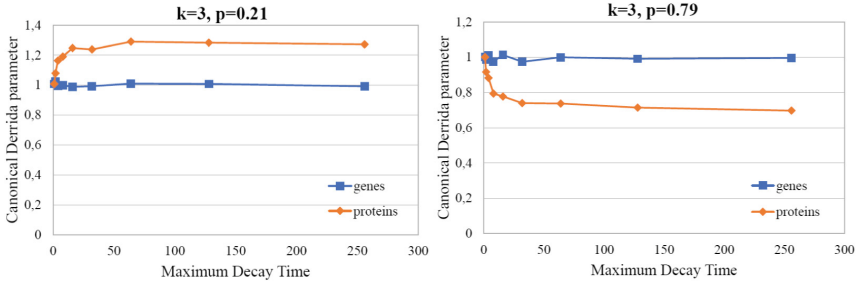


Fig. 3. Canonical Derrida parameter vs MDT, case $k = 3$: left $p = 0.21$, right $p = 0.79$. The two curves refer to the G-node and to the P-node subnetworks, subject to a P-node perturbation

ordered region; this happens for the case with high bias, while the Derrida parameters becomes larger than one in the low-bias case.

This behaviour may seem surprising (but see the comments in Sect. 5), therefore it is interesting to consider also the extended Derrida parameter described in Sect. 3. The results are shown in Figs. 4 and 5.

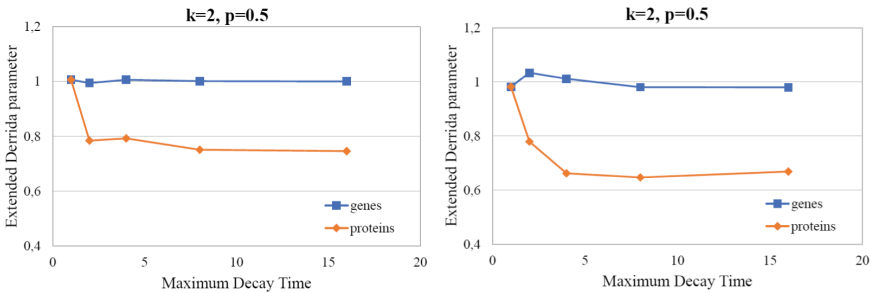


Fig. 4. Extended Derrida parameter vs maximum decay time for the case $k = 2$, $p = 0.5$; left $T_{ev} = 1$, right $T_{ev} = 3$. The two curves refer to the G-node and to the P-node subnetworks, subject to a P-node perturbation

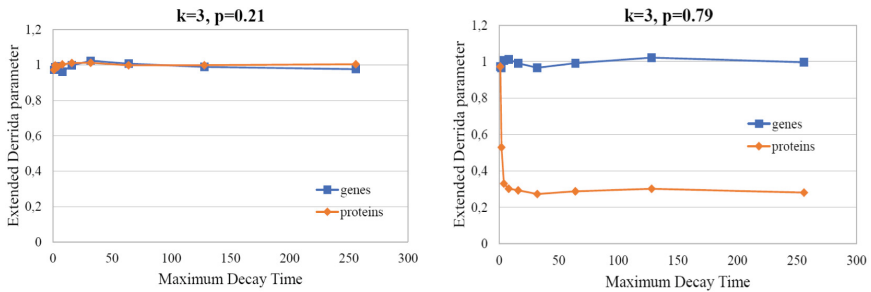


Fig. 5. Extended Derrida parameter vs maximum decay time for the case $k = 3$; Left $p = 0.21$, right $p = 0.79$. In both cases $T_{ev} = 3$. The two curves refer to the G-node and to the P-node subnetworks, subject to a P-node perturbation

Note that, while the G subnetwork remains critical, the behaviour of the P subnetwork is different from that of the canonical Derrida parameter. In the $k = 2$ case, it is more ordered ($\lambda < 1$ even for values of MDT slightly larger than 1) while it was critical in Fig. 2. In the $k = 3$, low-bias case the network is critical, while it was supercritical in Fig. 3. Only in the case of $k = 3$ with low bias the two behaviours are at least qualitatively the same. It should also be observed that the length of the time window T_{ev} may affect the outcomes: for example, by choosing it equal to one in the same case as that of Fig. 5 left, one would have concluded that the P subnetwork is slightly supercritical (data not shown here).

In order to complete the description of the model behaviours, let us now consider the results that have been obtained by perturbing the gene subnetwork (recall that all the previous ones referred to perturbations of P nodes). As it can be seen from Fig. 6 below, in all the cases both subnetworks are ordered even for values of MDT larger than 1.

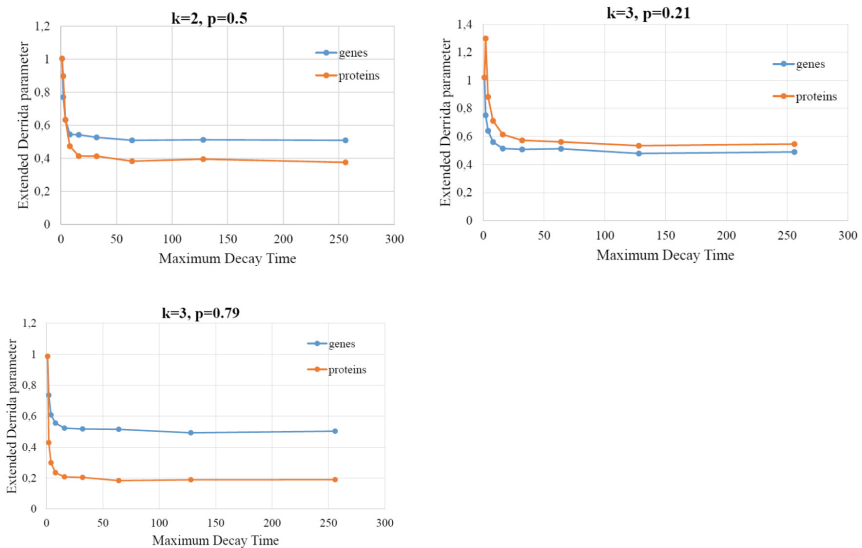


Fig. 6. Extended Derrida parameter vs maximum decay time for the cases $k = 2$ and $p = 0.5$, $k = 3$ and $p = 0.21$, $k = 3$ and $p = 0.79$. In all cases $T_{ev} = 1$. The curves refer to the G-node and to the P-node subnetworks, subject to a G-node perturbation

The dynamical regimes of GPBNs have been analysed so far by using canonical or modified Derrida methods, i.e. the discrete analogues of Lyapunov exponents. A major interest concerns the robustness of networks of this kind, and in order to characterize this property a different measure, independent of T_{ev} or of any similar parameter, is given by the fraction of perturbations that, starting from an attractor cycle, end in the same attractor.

These data are shown in Fig. 7. As it is expected, the fraction of perturbations that fall back onto the initial attractor decreases as the intensity of the perturbation increases. This fraction increases when a memory term is added and, like in the other cases described above, the effect is observed for small values of the maximum decay time, while further increases of MDT do not lead to any appreciable change.

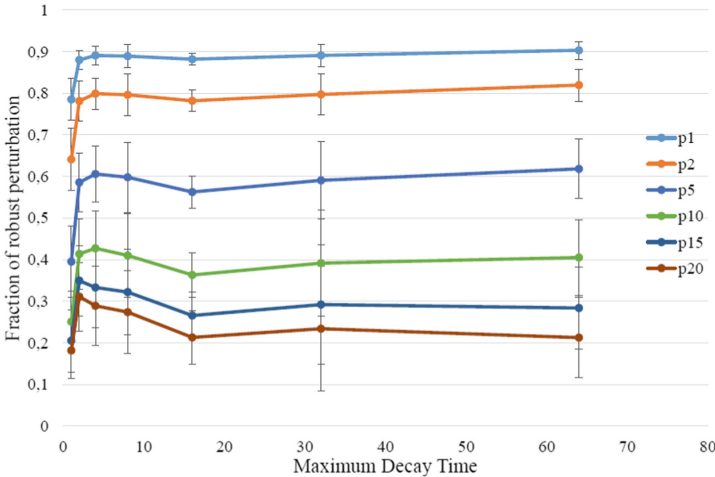


Fig. 7. The fraction of perturbations that came back to the starting attractor by varying MDT, if perturbing 1, 2, 5, 10, 15 or 20 P-nodes. Each point is the average of 50 different systems with 100 GP nodes: in each system the attractors are identified by using 100 random initial conditions; all states of the so sampled attractors are perturbed. In these experiments, we considered the same decay time for each P node.

5 Conclusion

The GPBN model of genetic regulatory systems maintains the abstraction level of the RBN framework and at the same time allows an explicit modelling of time delay effects.

It is of course extremely interesting to compare abstract-level models with real-world data. It has indeed been possible to show that RBNs can properly describe the distribution of perturbations in gene expression levels induced by single knock-outs in *S. Cerevisiae* [15, 16]. However, the techniques used for this purpose do not allow one to test the behaviour of the model when the perturbation affects several genes at the same time – a situation that is much more frequently encountered in experiments, like those related to the effects of drugs or contaminants. In these cases the comparison of model behaviour and experimental data should concern the time behaviour of the perturbation after the initial shock, but time-course data cannot be properly compared to RBNs because of their unrealistic synchronous updating. On the contrary, the introduction of memory terms in GPBNs should make it possible to deal also with

time-course data following a multiple initial perturbation, thus greatly increasing the wealth of experimental data available for testing the appropriateness of the abstract framework.

The kind of memory that has been introduced has different effects in case of information transmission from G to P nodes or from P to G nodes, and pose some interesting questions about the correct way of measuring of the system dynamical regimes through Derrida-like procedures. Anyway, the robustness of the system's attractors can constitute a sort of global measure related to its general "degree of order". In the future it will be interesting to analyse a Derrida parameter modified in a way different from those of Sect. 4, i.e. computed by allowing as initial states only those that belong to an attractor.

In order to understand the behaviour of the GPBN model when P nodes are perturbed, it will be interesting to consider separately the effects of up and down perturbations. Indeed, the impacts of "up" and "down" perturbations of P nodes are likely to have different intensities. The effect of a "down" perturbation, i.e. the disappearance of a protein, should typically die out quite rapidly, as the rest of the nodes resynthesize that protein. On the other hand, the impact of an "up" perturbation is likely to last longer, i.e. for a number of steps equal to its phase. Investigating the effects of the two types of perturbations by canonical and modified Derrida parameters may therefore provide important clues about the properties of the model.

References

1. Aldana, M., Coppersmith, S., Kadanoff, L.P.: Boolean dynamics with random couplings. In: Kaplan, E., Marsden, J.E., Sreenivasan, K.R. (eds.) *Perspectives and Problems in Nonlinear Science*, pp. 23–89. Springer, New York (2003). https://doi.org/10.1007/978-0-387-21789-5_2
2. Bartel, D.P.: MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell* **116**, 281–297 (2004)
3. Bartel, D.P.: MicroRNAs: target recognition and regulatory functions. *Cell* **136**(2), 215–233 (2009)
4. Campioli, D., Villani, M., Poli, I., Serra, R.: Dynamical stability in random Boolean networks In: Apolloni, B., Bassis, S., Esposito, A., Morabito, F.C. (eds.) *Frontiers in Artificial Intelligence and Applications, WIRN*, vol. 234, pp. 120–128. IOS Press, Amsterdam (2011)
5. Darabos, C., Giacobini, M., Tomassini, M.: Generalized Boolean networks: how spatial and temporal choices influence their dynamics. In: *Handbook of Research on Computational Methodologies in Gene Regulatory Networks*, pp. 429–449. IGI Global, Hershey (2010)
6. Derrida, B., Pomeau, Y.: Random networks of automata: a simple annealed approximation. *Europhys. Lett.* **1**(2), 45–49 (1986)
7. Derrida, B., Weisbuch, G.: Evolution of overlaps between configurations in random Boolean networks. *J. Phys.* **47**, 1297–1303 (1986)
8. Di Stefano, M.L., Villani, M., La Rocca, L., Kauffman, S.A., Serra, R.: Dynamically critical systems and power-law distributions: avalanches revisited. In: Rossi, F., Mavelli, F., Stano, P., Caivano, D. (eds.) *WIVACE 2015. CCIS*, vol. 587, pp. 29–39. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-32695-5_3

9. Graudenzi, A., Serra, R., Villani, M., Damiani, C., Colacci, A., Kauffman, S.A.: Dynamical properties of a Boolean model of gene regulatory network with memory. *J. Comput. Biol.* **18**, 1291–1303 (2011)
10. Graudenzi, A., Serra, R.: A new model of genetic network: the gene-protein network. In: Serra, R., Poli, I., Villani, M. (eds.) *Artificial Life and Evolutionary Computation*, pp. 283–291 (2009)
11. Graudenzi, A., Serra, R., Villani, M., Colacci, A., Kauffman, S.A.: Robustness analysis of a Boolean model of gene regulatory network with memory. *J. Comput. Biol.* **18**(4), 559–577 (2011)
12. Kauffman, S.A.: Homeostasis and differentiation in random genetic control networks. *Nature* **224**, 177–178 (1969)
13. Kauffman, S.A.: *The Origins of Order: Self Organization and Selection in Evolution*. Oxford University Press, Oxford (1993)
14. Kauffman, S.A.: *At Home in the Universe*. Oxford University Press, New York (1995)
15. Serra, R., Villani, M., Semeria, A.: Genetic network models and statistical properties of gene expression data in knock-out experiments. *J. Theor. Biol.* **227**, 149–157 (2004)
16. Serra, R., Villani, M., Graudenzi, A., Kauffman, S.A.: Why a simple model of genetic regulatory networks describes the distribution of avalanches in gene expression data. *J. Theor. Biol.* **246**, 449–460 (2007)
17. Shmulevich, I., Kauffman, S.A., Aldana, M.: Eukaryotic cells are dynamically ordered or critical but not chaotic. *Proc. Natl. Acad. Sci. U. S. A.* **102**, 13439–13444 (2005)
18. Villani, M., Campioli, D., Damiani, C., Roli, A., Filisetti, A., Serra, R.: Dynamical regimes in non-ergodic random Boolean networks. *Nat. Comput.* **16**(2), 353–363 (2017)
19. Darabos, C., Tomassini, M., Giacobini, M.: Dynamics of unperturbed and noisy generalized Boolean networks. *J. Theor. Biol.* **260**(4), 531–544 (2009)
20. Roli, A., Villani, M., Filisetti, A., Serra, R.: Dynamical criticality: overview and open questions. *J. Syst. Sci. Complex* **30**, 1–17 (2017)



Simulating Populations of Protocells with Uneven Division

Martina Musa¹, Marco Villani^{1,2}(✉), and Roberto Serra^{1,2}

¹ Department of Physics, Informatics and Mathematics,
University of Modena and Reggio Emilia, Modena, Italy
{marco.villani, rserra}@unimore.it

² European Centre for Living Technology, Ca' Foscari University, Venice, Italy

Abstract. Protocells should be similar to present-day biological cells, but much simpler. They are believed to have played a key role in the origin of life, and they may also be the basis of a new technology with tremendous opportunities. In this work we study the effect of uneven division processes on the synchronization of the duplication rates of protocells' membrane and internal materials.

Keywords: Protocell · Protocell populations · Models · Synchronization
Replicators

1 Introduction

Protocells should be similar to present-day biological cells, but much simpler [1, 2]. They are believed to have played a key role in the origin of life, and they may also be the basis of a new technology with tremendous opportunities (see e.g. the books [1, 3] and further references quoted therein). Among various candidate protocell architectures, those based on lipid vesicles are particularly promising since they can spontaneously undergo fission, giving rise to two daughter protocells. Protocells should also contain a self-replicating set of molecules (the “replicators”): their composition should also affect the growth and replication rate of the container, so that some kind of competition can take place between cells with different chemical compositions [4].

The simplest case is that of even division, where a vesicle splits into two identical daughter cells. In order to assure sustainable growth of a population of protocells, it is necessary that the duplication rate of the replicators be equal to that of the lipid container. It has been shown in a series of papers [5–10] that this synchronization takes spontaneously place, generation after generation, under a wide set of hypotheses concerning the protocell architecture and the kinetic equations for the replicators, and that it is robust with respect to random fluctuations. This is indeed a beautiful example of dynamical self-organization.

It has however also been observed that there are other ways in which lipid vesicles can divide. In this paper we consider a case (inspired by the “budding” processes [11]) where the vesicle splits in two daughter vesicles of different size, a “large” one and a “small” one. Other types of division, like e.g. those due to extrusion processes, might also be investigated. We suppose that, when a critical size has been reached, the protocell

splits into two: the daughters will inherit a fraction of both the lipid container and the replicators. In this work also consider the important situations of (noisy) uneven division and of not constant splitting thresholds.

The paper is organized as follows. In Sect. 2 we introduce the investigated protocell models and discuss the uneven division process. In Sect. 3 we present the experimental setting and the simulations, and discuss the simulations main results. Finally, in Sect. 4 we summarize the main paper results.

2 The Protocell Model

2.1 The Protocell Model

As anticipated, several different protocell “architectures” have been suggested [1, 12–14]. Many architectures are based upon lipid vesicles, where an aqueous internal environment is separated from the external water phase by a lipid bilayer, similar to those of existing biological cells. Vesicles form spontaneously under appropriate conditions, and it is known that they are able to split giving rise to two (or more) daughter cells [15, 16]. The different architectures are based on different hypotheses about the chemical composition of the protogenetic material (e.g., nucleic acids, or polypeptides, or even lipids themselves) and about the place where the action, i.e., duplication of genetic molecules and growth of the lipid container, takes place (in the internal environment, in the membrane, at the interface, or some combinations of the two) [13].

One might therefore be tempted to guess that no unified treatment is possible, however this turns out not to be the case: indeed it has been shown that at least the problem of synchronization lends itself to be dealt with using abstract models of quite broad applicability [3].

In this paper we consider the case where there is a single replicator, represented by some chemical species, and let its quantity (i.e., number of moles) be denoted by X . Let also C be the total quantity of “container” (i.e., lipid membrane forming vesicles or micelles) and V its volume, which is equal to C/ρ (where ρ is the density, assumed to be constant).

We assume that the X -molecule favors the formation of the container materials (as for example in [17, 18]). Some models imagine that only the X -molecule fraction near the external surface is effective in doing so, since the container precursors are found outside the protocell; other models envisage that these materials could pass through the membrane allowing in such a way an active role also to the internal X -molecule fraction. In [8] we demonstrate that these frameworks actually show equivalent behaviors: so, in this paper we follow the design that suppose permeable membranes and inner X -molecule materials.¹

So the catalytic activity of the X -molecules favor the growth of the lipid container, which provides in turn the physical conditions appropriate for the replication of the X -

¹ Note that even in this case it is possible that not all the internal X -molecules be active in supporting the container building; however, in [19] we show that also this difference does not significantly affect the process leading to the synchronization of the X -molecules and container reproduction rates.

molecules, without being however a proper catalyst. Because of its effect on the lipid container and of its ability in maintaining its presence during generations, the chemical species X loosely acts as a sort of “genetic material”, and we could refer to it with the term “genetic memory molecule”, or GMM for short.

If we follow the assumptions already used and discussed in [10], namely:

1. spontaneous container formation is negligible, so that only the catalyzed term matters
2. the precursors (both of container and X -molecule) are buffered
3. the membrane vesicle is thin, so the volume of the lipid membrane (and as consequence the amount of container C) is approximately proportional to its surface
4. diffusion is very fast, so in each phase the concentrations can be assumed to be homogeneous
5. the protocell breaks into two identical daughter units when its container mass reaches a certain threshold²
6. the shape of the mother protocell, as well as those of her daughters, are all spherical³
7. the rate limiting step which may appear in the replicator kinetic equations does not play a significant role when the protocell is smaller than the division threshold.

The simplified equations for the total quantities of lipid container C and replicator X during the continuous growth phase from an initial condition to the critical lipid container mass θ become:

$$\begin{cases} \frac{dC}{dt} = \alpha C^{\beta-1} X \\ \frac{dX}{dt} = \eta C^{\beta-1} X \end{cases} \quad (1)$$

where α and η are two positive constants denoting respectively the rate of self-replication of genetic molecules and the container growth, and the shape factor β ranges between $2/3$ for a micelle and 1 for a very thin vesicle [8].

As it was proved in [10], in order to determine whether there is a synchronization in the asymptotic time limit, one can limit oneself to consider the $\beta = 1$ case. The final result does not depend on β , while of course this parameter affects the speed with which it is approached: this is essentially a non-linear rescaling of time, useful to simplify the analysis. With this simplification, the basic equations (which are valid between two successive divisions) are then:

² The dropping of this hypothesis is one of the topics of this paper.

³ This assumption is reasonable if we suppose that the flow of water is “fast” enough to allow us to consider the protocell as turgid, on the time scale of interest [20]. This implies that we do not describe here in detail the breakup of a vesicle into two, which certainly requires consideration of shape changes – that are supposed to be fast and to fall below the time scale of the relevant phenomena that the model describes. Moreover, we do not take explicitly into account osmotic effects (as for example in [21]) that might be relevant in the case of hypertonic or hypotonic environments.

$$\begin{cases} \frac{dC}{dt} = \alpha X \\ \frac{dX}{dt} = \eta X \end{cases} \quad (2)$$

2.2 Uneven Division

As anticipated, we assume that a protocell splits into two daughters when its membrane reaches a certain critical mass, in the following indicated as θ . After splitting, one of the daughter cells inherits a fraction ω of the lipid container, while the other one inherits $1 - \omega$; the same happens to the GMM chemical species that are diluted in the membrane, if any.

In this paper we assume that the shape of the mother protocell, as well as those of her daughters, are all spherical. As it has been discussed elsewhere [3], this implies that some part of the mother's internal aqueous environment is lost in fission - and the same holds for the GMMs that are there diluted. So, only a part of these GMMs part is shared between the two daughters, which inherit $L\omega$ and $L(1 - \omega)$ respectively, L being the fraction of GMMs that is not lost in the bulk.⁴

This fraction can be calculated by simple geometric reasoning, if the concentration of the replicators is uniform in the internal water phase. Indeed, if r is the protocell radius, the surface and the volume of a protocell are respectively $S = 4\pi r^2$ and $V = 4\pi r^3/3$, and the dependence of volume from the surface is:

$$V = \frac{4}{3}\pi r^3 = \frac{4}{3}\pi \left(\frac{S}{4\pi}\right)^{\frac{3}{2}} = \frac{4}{3 \cdot 4\pi \cdot 2\sqrt{\pi}}\pi S^{\frac{3}{2}} = \frac{S^{\frac{3}{2}}}{6\sqrt{\pi}} \quad (3)$$

The sum of the volumes of the two daughters V_F therefore is:

$$V_F = \frac{\omega^{\frac{3}{2}}S^{\frac{3}{2}}}{6\sqrt{\pi}} + \frac{(1 - \omega)^{\frac{3}{2}}S^{\frac{3}{2}}}{6\sqrt{\pi}} = \frac{S^{\frac{3}{2}}}{6\sqrt{\pi}} \left(\omega^{\frac{3}{2}} + (1 - \omega)^{\frac{3}{2}}\right) = V_{mother} \left(\omega^{\frac{3}{2}} + (1 - \omega)^{\frac{3}{2}}\right) \quad (4)$$

Consequently, the fraction of the mother protocell volume that is not lost in the bulk is:

$$L = \omega^{\frac{3}{2}} + (1 - \omega)^{\frac{3}{2}} \quad (5)$$

This is also the fraction of GMMs of the initial protocell shared between the two daughters, in case of these GMMs are diluted in the mother's internal aqueous environments.

These rules determine the initial conditions of the two daughter cells at the next generation. The small one will need a longer time to reach the critical size and to undergo fission, while the larger one will be faster.

⁴ Obviously, $L = 1$ in case of the GMMs are diluted in the membrane.

The continuous growth described by Eq. 2, starting from an initial condition where $C = C_i$ and $X = x_i$ up to the time $T = T_{div}$ when $C = \theta$ (i.e. when splitting takes place) can be analytically determined to be

$$\begin{cases} T_{div} = \frac{\ln\left(\frac{\eta(\theta - C_i)}{\alpha x_i} + 1\right)}{\eta} \\ x_{div} = \frac{\eta(\theta - C_i)}{\alpha} + x_i \end{cases} \quad (6)$$

where x_{div} is the quantity of replicator at splitting time T_{div} .

Of course, in case of uniform and regular process of division (each progenitor regularly dividing into two descendants of size ω and $1-\omega$) a large daughter cell will also give rise to another large one, etc. This ‘‘pure lineage’’ of large cells will tend to synchronize, in a way similar to the case of even division. The same will happen for the pure lineage of small cells, although with a different division frequency. Therefore, for a pure ω lineage the following rule holds:

$$\begin{cases} T_{div} = \frac{\ln\left(\frac{1}{L\omega}\right)}{\eta} \\ x_{div} = \frac{\eta\theta}{\alpha} \frac{1-\omega}{1-L\omega} \end{cases} \quad (7)$$

A similar rule holds for the pure ‘‘ $1-\omega$ ’’ lineage, by substituting ω with ‘‘ $1-\omega$ ’’. Finally, we can compute the difference between the asymptotic division times and the GMMs’ quantities at division time of these pure lineages:

$$\begin{cases} \Delta T_{div} = \frac{\ln\left(\frac{(1-\omega)}{\omega}\right)}{\eta} \\ \Delta x_{div} = \frac{\eta\theta}{\alpha} \left(\frac{(L-1)(2\omega-1)}{1-L+L^2\omega-L^2\omega^2} \right) \end{cases} \quad (8)$$

Note that (i) the difference between the asymptotic division times does not depend upon L and that (ii) in case of even division these differences are equal to zero (only one lineage is present).

3 Population of Protocells

In the previous section, we derived the rules to compute the asymptotic division time and the GMMs’ quantity at division time of the protocell pure lineages.

However, as generations increase, the fraction of cells belonging to the pure lineages declines, and most cells have both large and small cells among their ancestors. Indeed, after k generations the large cells will be k , while the total number of cells will be 2^k , so the fraction of pure lineage cells will vanish in the long k limit.

An interesting question then concerns the distribution of division times after several generations: will there be, on average, a uniform distribution of fission events in time, or will there be some pace, at population level, in the fission processes?

We have therefore simulated the growth of such populations of protocells, originated from a single protocell. Because of computational limits (which anyway mimic real physical constraints) when the population size reach its maximum value each division implies the substitution of an already “born” protocell (stable population phase).

These substitutions are random, so the data are noisy. However, an interesting observation is that the data concerning both the fission intervals and the values of the replicators before fission are divided in two groups and they do not become homogeneous (Fig. 1). As expected, the division time is smaller in the case of the larger initial vesicles with a larger initial quantity of replicators (Fig. 1b). On the other hand, the smaller vesicles, with longer division times, synthesize a larger final quantity of

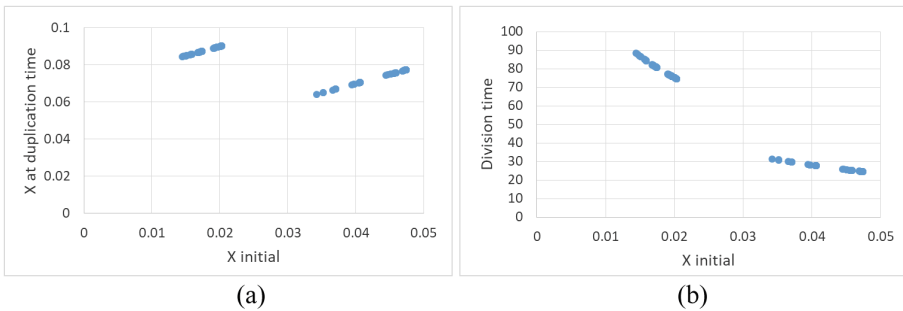


Fig. 1. (a) Quantity of replicators before division and (b) division time (i.e. the interval between two successive divisions) vs initial quantity of replicators.

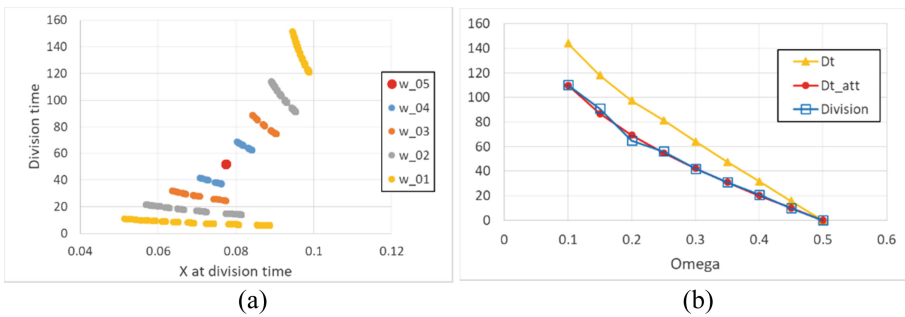


Fig. 2. (a) Populations of protocells with uneven fission exhibit two stable subgroups: the more uneven is the fission, the more distant are the characteristics of the subgroups. (b) Difference between the observed and the “theoretical” values for the maximum difference between division times, vs ω . Dt is the measured maximum distance in division times, Dt_{att} is the “theoretical distance” defined in the text, Division is the width of the zero-level plateau, whose some instances are visible in Fig. 3a and c.

replicators (Fig. 1a). The bimodality of the distribution of division times and of the quantity of replicators at division time can also be directly observed in Fig. 3, which shows their (stable) probability distribution for different ω . Therefore, after the growth phase the population of protocells shows the presence of two subgroups, composed respectively by protocells with relatively long lifespan that divide with high concentrations of GMMs, and protocells with relatively short lifespan that divide with low concentrations of GMMs. Each protocell divides into two (possibly uneven) daughters, but the two subpopulation are stable – in the sense that the cardinality of each group does not change in time.

It is also possible to analyze the difference between division times as a function of ω (remember that, given the geometrical hypotheses, ω determines also the fraction L of replicators that are not lost). Let us define the “theoretical distance” between division times as the difference between the division times of the two pure lineages, determined

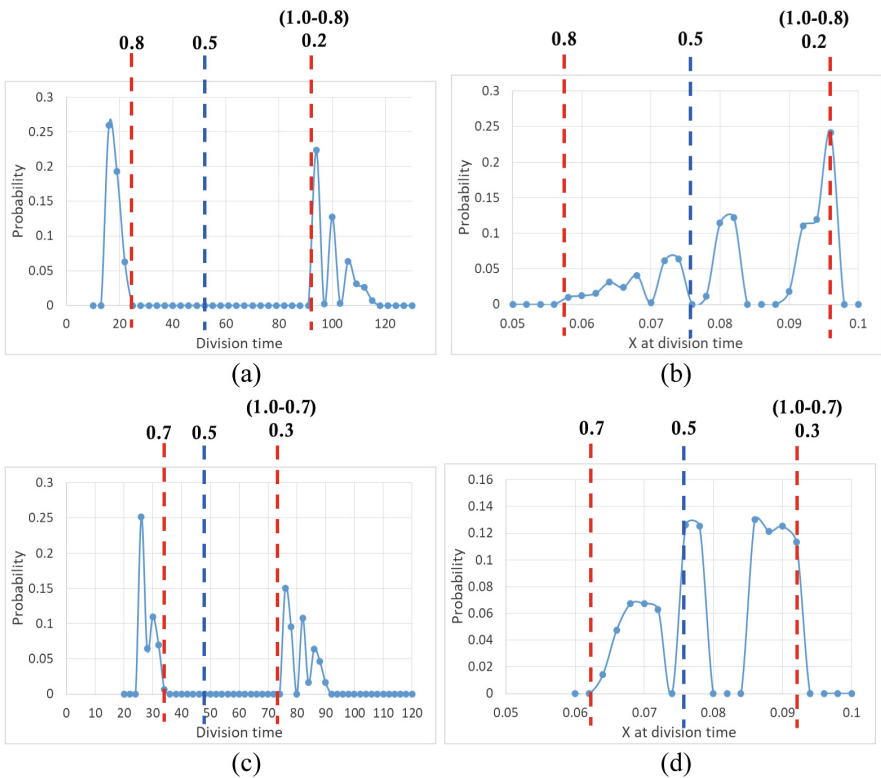


Fig. 3. Probability distribution of division times (a), (c) and probability distribution of X quantities at division time (b), (d) observed in simulations; the first row refers to $\omega = 0.2$, whereas the second row refers to $\omega = 0.3$. Red vertical lines indicate the division times and the X quantities at division time of the asymptotic pure lineages, blue vertical lines indicate the same for even division. Note that the division times of the pure lineages precisely individuates the extremities of the empty space dividing the two protocell subgroups, whereas their X quantities at division time embrace the distribution of the protocell populations. (Color figure online)

analytically in Eq. 8. Surprisingly enough, in the case of uneven division one sometimes observes that the maximum of the actual difference between division times typically exceeds this value, as shown in Fig. 2a. Indeed, the theoretical distance closely approximates the zero-level plateau (some instances are visible in Fig. 3a and c).

Till now protocells divide in two “new” individual each one inheriting respectively a fraction equal to $L\omega$ and $L\cdot(1-\omega)$ of the progenitor’s materials. However, this is a very idealized situation: real splitting processes are noisy (or even very noisy), and each

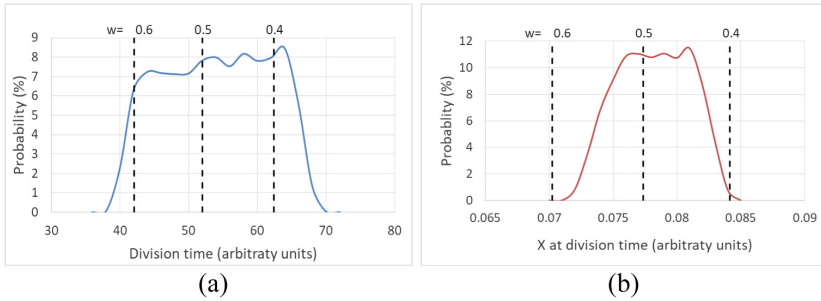


Fig. 4. Probability distribution of division times (a) and probability distribution of X quantities at division time (b) observed in simulations, when at division time each protocell divides in two “new” protocells inheriting respectively a fraction equal to $L\omega'$ and $L\cdot(1-\omega')$ of the progenitor’s materials, ω' being randomly extracted from a uniform distribution spanning within the range $[0.4, 0.5]$ (a process that simulates a noisy even division). The vertical lines indicate the division times and the X quantities at division time of pure lineages with ω respectively equal to 0.4 and 0.6 (the complementary size of a daughter protocell with $\omega = 0.4$).

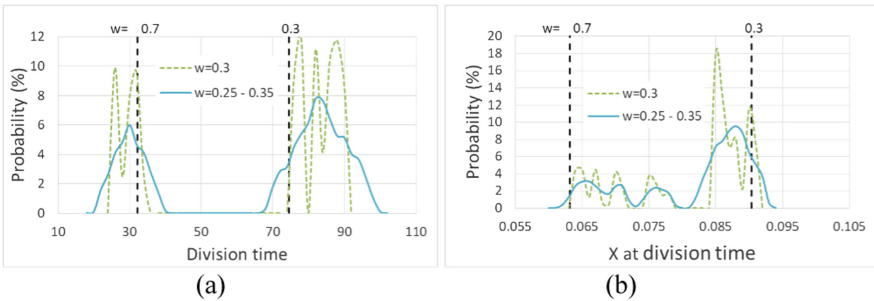


Fig. 5. Probability distribution of division times (a) and probability distribution of X quantities at division time (b) observed in simulations, when at division time each protocell divides in two “new” protocells inheriting respectively a fraction equal to $L\omega'$ and $L\cdot(1-\omega')$ of the progenitor’s materials, ω' being randomly extracted from a uniform distribution spanning within the range $[0.25, 0.3]$ (a process that simulates a noisy uneven division with $\omega = 0.3$). The vertical lines indicate the division times and the X quantities at division time of pure lineages with ω respectively equal to 0.3 and 0.7 (the complementary size of a daughter protocell with $\omega = 0.3$).

division could (i) happen at thresholds more or less distant from θ and/or (ii) giving birth to slightly different descendants.

The effect of adding noise to even division is that of “blurring” the delta distributions of the asymptotic division times and of the X quantities at division time (Fig. 4); a similar effect is observed in uneven divisions, where it is still possible however to re-observe the presence of the two previously individuated protocell sub-populations (Fig. 5).

Interestingly, a very “disordered” division process leads to an asymmetric distribution of division times (with a small fraction of protocell owning very long lifespans), that corresponds to a very sparse X quantities at division times distribution. Note that the presence of very long lifespans allows the formation of protocells with relatively high final concentrations (Fig. 6).

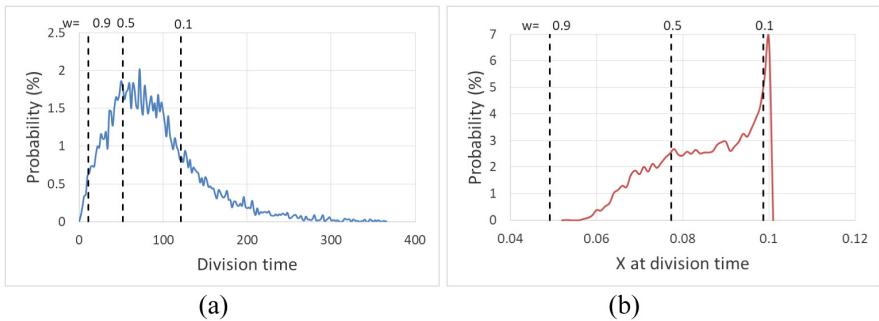


Fig. 6. Probability distribution of division times (a) and probability distribution of X quantities at division time (b) observed in simulations, when at division time each protocell divides in two “new” protocells inheriting respectively a fraction equal to $L\omega'$ and $L\cdot(1-\omega')$ of the progenitor’s materials, ω' being randomly extracted from a uniform distribution spanning within the range $[0.01, 0.5]$ (a process that simulates a very noisy uneven division). The vertical lines indicate the division times and the X quantities at division time of pure lineages with ω respectively equal to 0.5, 0.9 and 0.1 (the complementary size of a daughter protocell with $\omega = 0.9$).

The effect of splitting events occurring at not constant protocell size (simulated by adding noise to θ , the membrane threshold quantity that induces the protocells instabilities leading to the protocell division) is similar to the ones just shown for noisy uneven division. A random size of splitting makes more “fuzzy” the delta distributions of the asymptotic division times and of the X quantities at division time (Fig. 7); very high noise levels possibly override the presence of the two previously individuated protocell subpopulations, at least of one of the observed variables (Fig. 7d).

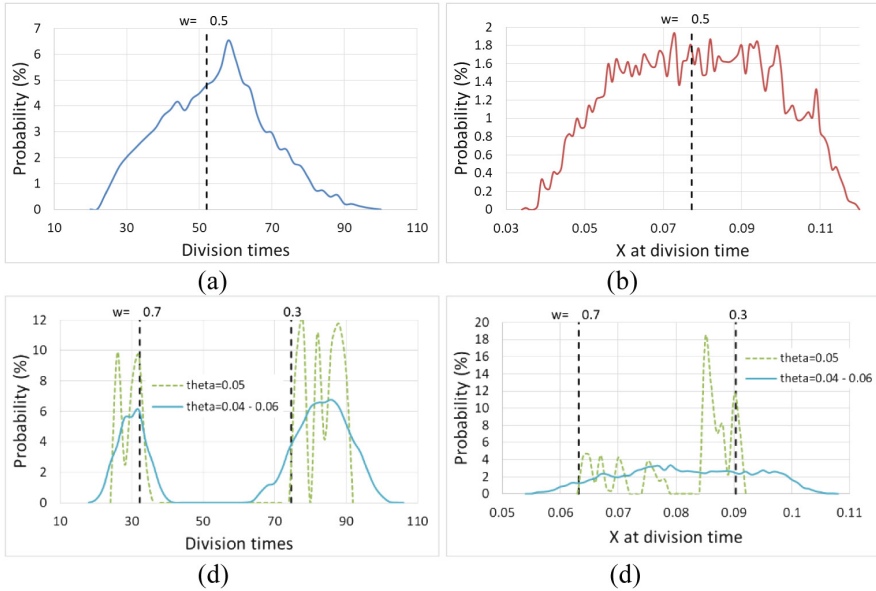


Fig. 7. Probability distribution of division times (a), (c) and probability distribution of X quantities at division time (b), (d) observed in simulations, for even division (first row) and uneven division (second row, uneven division with $\omega = 0.3$), in case of the membrane threshold quantity that induces the protocells instabilities leading to the protocell division is not constant. The vertical lines indicate the division times and the X quantities at division time of pure lineages with ω respectively equal to 0.5, 0.7 and 0.3 (the complementary size of a daughter protocell with $\omega = 0.7$).

4 Conclusions

Protocells should be similar to, but much simpler than biological cells. Protocell populations do not yet exist and mathematical models are therefore extremely important to address the key questions concerning their synthesis and behavior. Different protocell architectures have been proposed so, due to uncertainties about the details, high-level abstract models like those that are presented in this paper are particularly relevant. In this context, the problem of synchronization plays a particularly relevant role: indeed, growth and evolution of a population of protocells require that reproduction of the whole protocell and replication of its “genetic memory molecules” take place at the same pace.

Despite the fact that only “pure lineage” streams of protocells can rigorously synchronize (that is, reproduce the same amount of materials at precise and regular time intervals), in this paper we show that the macroscopic output of the random superposition of thousands of these processes is the presence within the protocell population of stable distributions of the relevant protocell variables. In case of uneven division

these distributions become bimodal, highlighting in such a way the presence of two stable subpopulations, the macroscopic consequence of the fact that protocells, when divide, split into two not symmetric descendants.

Further works will explore the effects of changing the protocell architecture on the regularity or on the shape of this very interesting macroscopic output.

References

1. Rasmussen, S., Bedau, M.A., Chen, L., Deamer, D., Krakauer, D.C., Packard, N.H., Stadler, P.F. (eds.): *Protocells*. The MIT Press, Cambridge (2008)
2. Schrum, J.P., Zhu, T.F., Szostak, J.W.: The origins of cellular life. *Cold Spring Harb. Perspect. Biol.* **2**, a002212 (2010)
3. Serra, R., Villani, M.: A stochastic model of growing and dividing protocells. *Modelling Protocells*. UCS, pp. 105–147. Springer, Dordrecht (2017). https://doi.org/10.1007/978-94-024-1160-7_5
4. Smith, J.M., Szathmáry, E.: *The Major Transitions in Evolution*. W.H. Freeman Spektrum, Oxford (1995)
5. Serra, R.: The complex systems approach to protocells. In: Pizzuti, C., Spezzano, G. (eds.) *Advances in Artificial Life and Evolutionary Computation, WIVACE 2014*. Communications in Computer and Information Science, vol. 445, pp. 201–211. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12745-3_16
6. Villani, M., Filisetti, A., Graudenzi, A., Damiani, C., Carletti, T., Serra, R.: Growth and division in a dynamic protocell model. *Life* **4**, 837–864 (2014)
7. Filisetti, A., Serra, R., Carletti, T., Villani, M., Poli, I.: Non-linear protocell models: synchronization and chaos. *Europhys. J. B* **77**, 249–256 (2010)
8. Carletti, T., Serra, R., Poli, I., Villani, M., Filisetti, A.: Sufficient conditions for emergent synchronization in protocell models. *J. Theor. Biol.* **254**, 741–751 (2008)
9. Filisetti, A., Serra, R., Carletti, T., Poli, I., Villani, M.: Synchronization phenomena in protocell models. *BRL. Biophys. Rev. Lett.* **3**(1/2), 325–342 (2008)
10. Serra, R., Carletti, T., Poli, I.: Synchronization phenomena in surface reaction models of protocells. *Artif. Life* **13**, 1–16 (2007)
11. Svetina, S.: Vesicle budding and the origin of cellular life. *ChemPhysChem* **10**, 2769–2776 (2009)
12. Solé, R.V., Macía, J., Fellermann, H., Munteanu, A., Sardanyés, J., Valverde, S.: Models of protocell replication. In: Rasmussen, S., Bedau, M.A., Chen, L., Deamer, D., Krakauer, D. C., Packard, N.H., Stadler, P.F. (eds.) *Protocells*, pp. 213–231. The MIT Press, Cambridge (2008)
13. Ruiz-Mirazo, K., Briones, C., de la Escosura, A.: Prebiotic systems chemistry: new perspectives for the origins of life. *Chem. Rev.* **114**, 285–366 (2014)
14. Luisi, P.L., Ferri, F., Stano, P.: Approaches to semi-synthetic minimal cells: a review. *Naturwissenschaften* **93**, 1–13 (2006)
15. Luisi, P.L.: *The Emergence of Life: From Chemical Origins to Synthetic Biology*. Cambridge University Press, New York (2007)
16. Terasawa, H., Nishimura, K., Suzuki, H., Matsuura, T., Yomo, T.: Coupling of the fusion and budding of giant phospholipid vesicles containing macromolecules. *Proc. Natl. Acad. Sci.* **109**, 5942–5947 (2012)

17. Rasmussen, S., Chen, L., Stadler, B.M.R., Stadler, P.F.: Photo-organism kinetics: Evolutionary dynamics of lipid aggregates with genes and metabolism. *Orig. Life Evol. Biosph.* **34**, 171–180 (2004)
18. Rocheleau, T., Rasmussen, S., Nielsen, P.E., Jacobi, M.N., Ziock, H.: Emergence of protocellular growth laws. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **362**, 1841–1845 (2007)
19. Calvanese, G., Villani, M., Serra, R.: Synchronization in near-membrane reaction models of protocells. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016. CCIS*, vol. 708, pp. 167–178. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_15
20. Sacerdote, M.G., Szostak, J.W.: Semipermeable lipid bilayers exhibit diastereoselectivity favoring ribose. *Proc. Natl. Acad. Sci.* **102**, 6004–6008 (2005)
21. Mavelli, F., Ruiz-Mirazo, K.: Theoretical conditions for the stationary reproduction of model protocells. *Integr. Biol.* **5**, 324–341 (2013)



An Integrated Model Quantitatively Describing Metabolism, Growth and Cell Cycle in Budding Yeast

Pasquale Palumbo^{1,2(✉)}, Marco Vanoni^{1,3}, Federico Papa^{1,2},
Stefano Busti^{1,3}, Meike Wortel^{4,5}, Bas Teusink⁴,
and Lilia Alberghina^{1,3}

¹ SYSBIO Centre for Systems Biology, Milan, Italy

pasquale.palumbo@iasi.cnr.it,

{marco.vanoni, lilia.alberghina}@unimib.it

² Institute for System Analysis and Computer Science

“Antonio Ruberti” – CNR, Via dei Taurini 19, Rome, Italy

³ Department of Biotechnology and Biosciences, University of Milano-Bicocca,
Piazza Della Scienza 2, Milan, Italy

⁴ Systems Bioinformatics, VU University,

De Boelelaan 1087, 1081 HV Amsterdam, The Netherlands

⁵ Centre for Ecological and Evolutionary Synthesis (CEES), The Department
of Biosciences, University of Oslo, Blindernveien 31, 0371 Oslo, Norway

Abstract. Computational models are expected to increase understanding of how complex biological functions arise from the interactions of large numbers of gene products and biologically active low molecular weight molecules. Recent studies underline the need to develop quantitative models of the whole cell in order to tackle this challenge and to accelerate biological discoveries.

In this work we describe three major functions of a yeast cell: Metabolism, Growth and Cycle, through two coarse grain models, MeGro (Metabolism + Growth) and GroCy (Growth + Cycle). GroCy effectively recapitulates major phenotypic properties of cells grown in glucose and ethanol supplement media. MeGro can act as a parameter generator for GroCy. The resulting iMeGroCy integrated model can be used as a scaffold for molecularly detailed models of yeast functions.

Keywords: Computational models · Systems biology · Whole cell models

1 Introduction

Saccharomyces cerevisiae is a major eukaryotic model organism in both fundamental and applied research. Computational approaches are required to analyze, structure and integrate the ever-increasing data sets available for yeast. Ultimately, a dynamic, comprehensive computational model of *S. cerevisiae* should be the ambition: it would, in part, allow further improvement of industrial bioprocesses by extending the understanding presently possible by genome-scale metabolic model [1]. It also would allow translation of the methodologies to human cells, as it previously happened for

genome sequencing, functional analysis and interactomics, just to name a few fields in which yeast research has recently led the way [2].

The design rules followed in the construction of the pioneering *Mycoplasma* whole cell model [3] were to divide the functionality of the cell into modules, each modeled bottom-up for short enough periods of time to assume module independence. Simple translation of this approach to a eukaryote, even as simple as the unicellular budding yeast, may not be straightforward. In fact, in contrast to *Mycoplasma*, yeast has a compartmentalized cellular organization, a ten-fold larger genome [4], sophisticated nutritionally modulated sensing and differentiation pathways [5] and an asymmetric cell division that results in population heterogeneity in terms of size, age and cellular content of individual cells. Accordingly, the successful building of models of cells more complex than *Mycoplasma* may face significant challenges [6] and originate models that are difficult to structure and parametrize.

To deal with yeast complexity we developed a multi-level approach. In the following, we present an integrated coarse grain model of the basic functions of a yeast cell (metabolism, growth and cycle), investigate how they respond to availability of a major yeast nutrient, glucose, and discuss how the model can be used as a scaffold for molecularly detailed models of yeast functions.

2 The Metabolism and Growth Model (MeGro)

The Metabolism and Growth Model (MeGro) connects growth and metabolism in *S. cerevisiae*. Growth rate maximization forms a rational basis for explaining growth strategies (see e.g. [7] and references therein), since a faster growing unicellular microorganism will have higher evolutionary fitness than its competitors, producing more progeny per time unit in a given environment. So we considered a coarse grain representation of a yeast cell that maximizes its specific growth rate by allocating total protein synthesis capacity to different protein pools. MeGro - derived from the generic “self-replicator” model proposed in [7] for unicellular microorganisms - is conceived to highlight the common patterns connecting growth rate-dependent regulation of cell size, ribosomal content and metabolic efficiency in a cell. All the metabolic reaction rates, the kinetic parameters and the stoichiometry of the flux balance constraints in MeGro are suitably tuned for *S. cerevisiae* and only the relevant classes of enzymes and metabolites are considered.

MeGro accounts for five classes of proteins and five kinds of metabolites. The proteins with enzymatic activity (square blocks in the MeGro scheme of Fig. 1) are (i) the hexose transporters, ‘*hxt*’, (ii) the glycolytic enzymes, ‘*gly*’, (iii) the ribosomes, ‘*rib*’, (iv) the respiration and (v) fermentation pathways enzymes, ‘*resp*’ and ‘*ferm*’ respectively. Three kinds of metabolites are involved in metabolic conversions (green ovals in the MeGro scheme of Fig. 1): the (a) extracellular and (b) intracellular glucose, ‘*glc, ex*’, and ‘*glc, in*’ respectively, and (c) pyruvate, ‘*pyr*’; other two kinds of metabolites are involved in energy production/consumption: (d) *ATP* and (e) *ADP*. In the following we indicate with c_x , $x \in \{Prot, Met\}$, $Prot = \{hxt, gly, rib, resp, ferm\}$, $Met = \{glc, ex, glc, in, pyr, ATP, ADP\}$, the protein/metabolite concentrations, (mM), and with v_x , $x \in Prot$, the metabolite fluxes, (mM/h) catalyzed by a specific protein x .

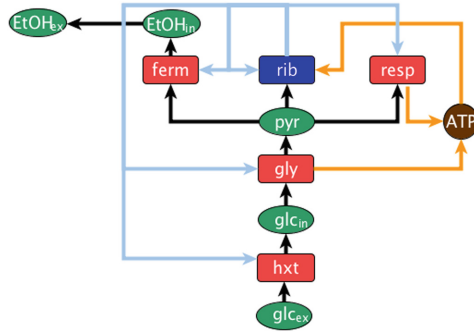


Fig. 1. Concept map of MeGro sub-module. (Color figure online)

MeGro captures resource allocation strategies, partitioning the investment into ribosomes in producing the different metabolic proteins. The accumulation of each protein pool is provided by a proper fraction α_x of the net ribosomal flux v_{rib} . Thus, in exponential growth conditions we have the following steady-state constraints (see also [7] for analytical details):

$$\lambda c_x - \alpha_x v_{rib} = 0, \quad x \in Prot \quad (1)$$

where λ (h^{-1}) is the specific growth rate and $\sum_{x \in Prot} \alpha_x = 1$, with $\alpha_x \geq 0$. Equation (1) refers to a steady state, each protein pool resulting from the balance - not explicitly modeled - of synthesis and degradation. Protein synthesis and degradation are instead explicitly modeled in the GroCy dynamical model, detailed in the next section.

The net dynamics of $c_{glc,in}$, c_{pyr} , c_{ATP} are determined by the combination of the fluxes of production and consumption:

$$dc_{glc,in}/dt = v_{hxt} - v_{gly}, \quad (2)$$

$$dc_{pyr}/dt = 2v_{gly} - v_{ferm} - v_{resp} - 600v_{rib}, \quad (3)$$

$$dc_{ATP}/dt = 2v_{gly} + 10v_{resp} - 2000v_{rib}, \quad (4)$$

providing steady-state constraints by imposing the derivatives equal to zero.

The total amount of ADP + ATP is constant, according to the following relationship

$$c_{ATP} + c_{ADP} = 1. \quad (5)$$

All protein and metabolite concentrations c_x are such that $c_x \geq 0$. All the metabolic conversions are catalysed by enzymes and the corresponding fluxes are modeled using the Michaelis-Menten formalism:

$$v_{hxt} = \frac{k_{cat,hxt}c_{glc,ex}c_{hxt}}{(c_{glc,ex} + k_{m,hxt})(1 + c_{glc,in}/k_{i,glui})}, \quad (6)$$

$$v_{gly} = \frac{k_{cat,gly}c_{ADP}c_{glc,in}c_{gly}}{(c_{ADP}c_{glc,in} + k_{m,gly}c_{ADP} + k_{m,ADP}c_{gly}c_{glc,in} + k_{gly}k_{m,ADP}c_{gly})(1 + c_{pyr}/k_{i,pyr})}, \quad (7)$$

$$v_{rib} = \frac{k_{cat,rib}c_{ATP}c_{pyr}c_{rib}}{c_{ATP}c_{pyr} + k_{m,rib}c_{ATP} + k_{m,ATP}c_{rib}c_{pyr} + k_{m,rib}k_{m,ATP}c_{rib}}, \quad (8)$$

$$v_{resp} = \frac{k_{cat,resp}c_{ADP}c_{pyr}c_{resp}}{c_{ADP}c_{pyr} + k_{m,resp}c_{ADP} + k_{m,ADP}c_{resp}c_{pyr} + k_{m,resp}k_{m,ADP}c_{resp}}, \quad (9)$$

$$v_{ferm} = \frac{k_{cat,ferm}c_{pyr}c_{ferm}}{c_{pyr} + k_{m,ferm}}. \quad (10)$$

(1) to (10) define the set of algebraic-differential equations of MeGro. The exponential growth rate λ is maximized as a function of the external glucose concentration $c_{glc,ex}$ (model input), with the fractions α_x as optimization variables, and subject to exponential growth constraints (1), flux balance constraints (derived from steady-state Eqs. (2–4)), feasible constraints (5) and Michaelis-Menten flux Eqs. (6–10).

The optimal set of λ and α_{hxt} , α_{gly} , α_{rib} , α_{resp} , α_{ferm} together with the proteins/metabolites concentrations and the protein fluxes provide a first level of MeGro outputs. A second level of cellular outcomes are computed by properly exploiting concentrations and fluxes. These are (i) the fermentative ratio F ,

$$F = v_{ferm} / (v_{ferm} + v_{resp}), \quad (11)$$

(ii) the ribosome-over-protein ratio ρ ,

$$\rho = c_{rib} / (600 (c_{hxt} + c_{gly} + c_{rib} + c_{resp} + c_{ferm})), \quad (12)$$

with proteins expressed in terms of number of polymerized amino acids, which explains the division by 600, the average number of polymerized amino acids per protein [8, 9] and (iii) the yield of ethanol $Y_{EtOH/glc}$,

$$Y_{EtOH/glc} = v_{ferm} / v_{hxt}. \quad (13)$$

If we leave F as an optimization variable, the model predicts that the cell behavior is fully respiratory for values of external glucose smaller than a critical value and then switches to purely fermentative for values of external glucose greater than the threshold. This model behavior (respiratory-to-fermentative switch) is independent of the setting of the model parameters (no threshold mechanism is artificially imposed), instead it is an emergent property of MeGro, with the model parameters allowing the tuning of the value of the external glucose threshold. Such a behavior is coherent with experimental results showing ethanol production only when the dilution rate exceeds a certain level, see e.g. [16].

MeGro can treat the fermentative ratio F as an input rather than an output, thus allowing the modeler to compute the optimal growth rate (as well as all the other model outputs) according to different values of F . Indeed, by properly exploiting the flux balance constraints (derived from steady state Eqs. (2–4)) and the fermentative ratio Definition (11), we can write v_{ferm} and v_{hxt} in terms of F and of the ribosomal flux v_{rib} :

$$v_{ferm} = (1.4 \times 10^3 F) / (1 + 10(1-F)) v_{rib}, \quad (14)$$

$$v_{hxt} = 10^3 \times (1 + 3(1-F)) / (1 + 10(1-F)) v_{rib}, \quad (15)$$

so that, according to the ethanol yield Definition (13) the fermentative ratio F is provided as a function of a given ethanol yield:

$$F = 20Y_{EtOH/glc} / (7 + 15Y_{EtOH/glc}). \quad (16)$$

This last equation will be exploited to feed MeGro with the fermentative ratio associated to experimental yield, Fig. 2A.

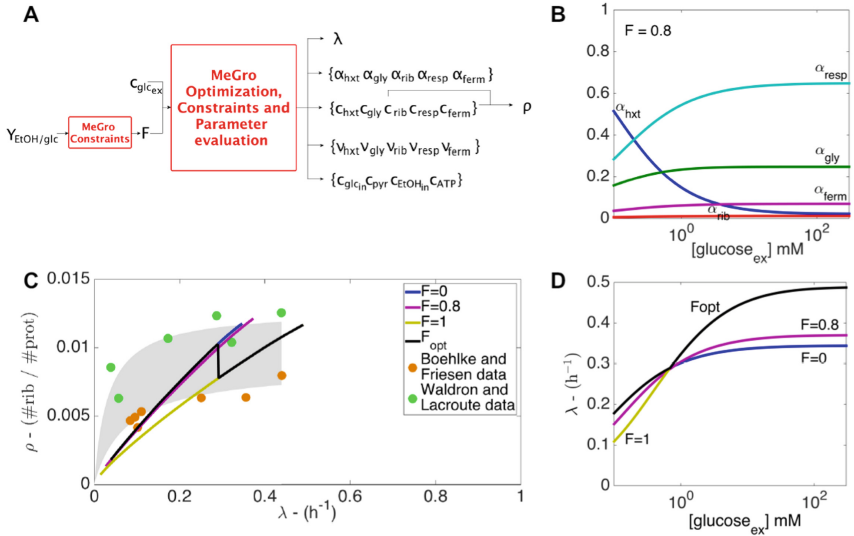


Fig. 2. AMeGro outcomes, when both the external glucose concentration and the fermentative ratio (i.e., the yield of ethanol) are exploited as model inputs. **B:** optimal fractions of ribosomal activity (α_j) engaged in the synthesis of the corresponding protein modules as functions of the external glucose concentration. **C:** MeGro optimal ribosome-over-protein ratio ρ as a function of MeGro optimal growth rate λ , with fixed fermentative ratio F (colored curves, F ranging in $[0, 1]$) and not fixed F (bold black line). MeGro simulations are compared to the experimental data redrawn from [10, 11]: in grey we highlight the region between Michaelis-Menten experimental data best fitting. **D:** MeGro optimal growth rate λ as a function of the external glucose concentration $C_{glc,ex}$, with fixed fermentative ratio F (colored curves, F ranging in $[0, 1]$) and not fixed F (bold black line). (Color figure online)

Figure 2B reports the steady-state protein fluxes for the different protein pools as a function of glucose concentration for a fermentative ratio $F = 0.8$. λ increases as a function of external glucose concentration following a saturation kinetics, whose parameters depend on the fermentative ratio F (Fig. 2D). Figure 2C shows the behavior of the ribosome-over-protein ratio ρ as a function of the external glucose concentration, at different fixed values of the fermentative ratio F . Model predictions (solid lines) are compared with two sets of experimental data (green and orange circles) from different yeast strains [10, 11], showing overall agreement between model predictions and experimental data. MeGro parameters can be found in Table 1. Most parameters are chosen by following the criteria developed in [12], with minor modifications, mostly related to the use of different units.

Table 1. MeGro parameters. Most parameters are chosen by following to the same criteria developed in [12], with minor modifications, mostly related to the use of different units. To determine the parameters $k_{cat,x}$ $x \in \{Prot\}$, we used experimental data on the specific growth rate of yeast cell populations growing in batch cultures at different glucose concentrations and literature data of yeast cells in chemostat. Since the maximal growth rate is reached by fully fermenting cells, we tuned $k_{cat,x}$ (except $k_{cat,resp}$) in order to fit our maximal experimental growth rate of 0.424 h^{-1} at a glucose concentration of 278 mM, obtaining a maximal growth rate of 0.48 h^{-1} for fully fermenting cells at saturating glucose concentrations. In order to tune $k_{cat,resp}$ we consider literature data reporting the growth rate at which the switch from respiration to fermentation occurs. According to such data, *S. cerevisiae* in a chemostat starts to produce ethanol at a dilution rate between 0.25 and 0.28 h^{-1} [13]. Then we set $k_{cat,resp}$ such that an equal growth rate of about 0.28 h^{-1} is achieved either using the fermentation pathway or the respiration pathway.

Parameter	Meas. unit	Value	Parameter	Meas. unit	Value
$k_{cat,hxt}$	h^{-1}	37492	k_m, hxt	mM	20
$k_{cat,gly}$	h^{-1}	4166	k_m, gly	mM	0.2
$k_{cat,rib}$	h^{-1}	670	$k_m, rib, k_i, pyr, k_i, glui$	mM	1
$k_{cat,resp}$	h^{-1}	99	$k_m, resp, k_m, ADPgly, k_m, ADPresp, k_m, ATPrib$	mM	0.5
$k_{cat,ferm}$	h^{-1}	6427	$k_m, ferm$	mM	5

3 The Growth and Cycle Model (GroCy)

In yeast, the critical cell size required to enter S phase (P_S) is modulated by nutrient availability [14]. It remains small and nearly constant when glucose is utilized through respiration. In contrast, P_S and hence average protein content increases as cells shift their metabolism towards fermentation [15]. Cells forced to ferment under slow-growing conditions show the same increase [16].

GroCy is composed by three modules (Fig. 3): (1) a dynamical cell growth model in which a set of ordinary differential equations describes dynamics of synthesis and degradation of ribosomes and proteins; (2) a molecular triggering mechanism that links cell growth and cell cycle. It exploits a set of ordinary differential equations which detail the dynamics of the growth-controlled activator Cdk1Cln3 and of its cognate

inhibitor Far1; (3) a cell cycle module, that consists of three consecutive timers (T_{1b} , T_2 and T_B) that describe the cycle progression after the triggering mechanism activates the first timer T_{1b} . The period that leads from the birth of the cell up to the time instant when the molecular machinery triggers the first timer is denoted by T_{1a} .

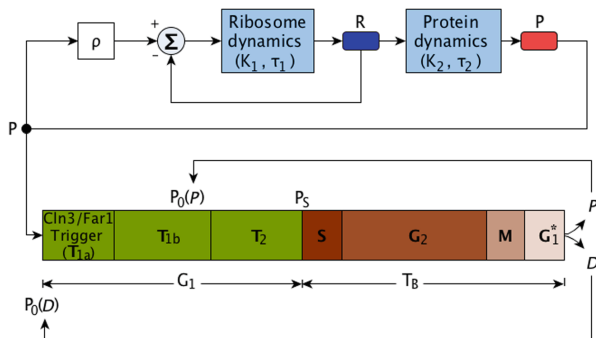


Fig. 3. Concept map of GroCy sub-module.

3.1 The Growth Module

The growth module deals with the ribosome content R , expressed as number of ribosomes per cell (rib), and the protein content P , expressed as number of polymerized amino acids per cell (aa), and is taken from [17] (where the reader can find the equations and the details which are below briefly recalled). Both ribosome and protein dynamics are described by the balance between production and degradation rates. Figure 4A shows the time course of the protein content and of the number of ribosomes for two different parameter settings: fast growth (2% glucose, solid line) or slow growth (ethanol, dashed line).

For each steady-state growth condition, the target ribosome/protein ratio ρ is an output of MeGro that can be directly fed into GroCy, providing the link between the two models. According to the model, when the ratio R/P is greater than ρ , then there is no ribosome production; otherwise, the ribosome production rate is proportional to the (positive) difference $\rho P - R$. Denoting with K_2, τ_2 , the average translational efficiency and the protein degradation time constant, respectively, it can be shown that, provided the exponential growth condition is satisfied, $\rho K_2 - 1/\tau_2 > 0$, the ratio R/P asymptotically converges to the value of parameter ρ . The exponential growth condition ensures that both ribosomes and proteins grow according to the same exponential law, with an exponential growth rate λ (min^{-1}) given by: $\lambda = \rho K_2 - 1/\tau_2$.

λ is not hard-wired in the model, but rather it is linked to the macromolecular composition and biosynthetic activity of the cells, a connection whose detection is made possible by the appropriate choice of the measurement units for ribosome and protein content, synthesis and degradation.

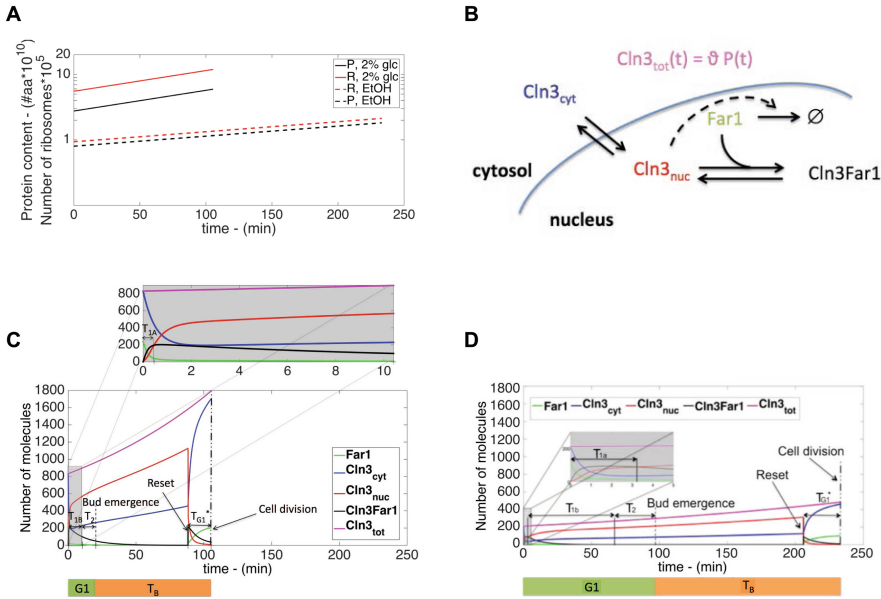


Fig. 4. A Time course of the protein content P and of the number of ribosomes R for fast (2% glucose, solid line) and slow growth conditions (ethanol, dashed line). **B:** schematic view of the interactions among the molecular players involved in the GroCy molecular triggering mechanism. **C-D:** time evolution of the players of the molecular triggering module, for fast (2% glucose, panel C) and slow (ethanol, panel D) growth conditions.

3.2 The Molecular Trigger Linking Growth to Cell Cycle Initiation

In budding yeast the entrance into S phase and budding starts when cells reach a critical cell size, thus connecting growth and cycle [18, 19]. Cln3 is an activator of S phase entrance, whose amount is proportional to the overall protein content, Eq. (20), therefore linking the growth and cycle modules. Despite some discordant results discussed in [20], we take that Cln3 accumulation is constant during G1. Cln3 production takes place in the cytoplasm. Cytoplasmic Cln3 is defined straightforwardly by Eq. (21). Nuclear volume is a constant fraction of total cell volume throughout the cycle [21].

The cyclin-dependent kinase inhibitor Far1 is involved in the cell size control mechanism in cycling cells by inhibiting Cln3 in early G1 [22, 23]. After mitosis, newly synthesized Far1 is endowed to each nucleus [24]. Ensuing Cln3 nuclear transport and accumulation allows overcoming of Far1 inhibition, which is made irreversible by Far1 degradation primed by the rising Cln3 activity [25, 26].

Here we use a simplified version of the equations used in [27] to model the molecular interplay between Cln3 and Far1, which we call the molecular triggering mechanism. Cdk1 – present in excess over its regulatory subunits – is implied, but not explicitly modeled (Fig. 4B). Cln3 transport in the nucleus and Cln3/Far1 interaction follow mass action kinetics. Far1 degradation is governed by rate η (min^{-1}) modeled

according to a Hill function (Eq. (21)), that increases from 0 to a high level η^* as soon as the free nuclear Cln3 exceeds Cln3Far1. nF is the Hill coefficient, modeling the steepness of the Hill function. Equation (19) accounts for reversible nucleocytoplasmic transport of Cln3 and interaction with Far1.

$$dCln3Far1/dt = (k_{on}/V_{nuc})Cln3_{nuc}Far1 - k_{off}Cln3Far1, \quad (17)$$

$$dFar1/dt = -(k_{on}/V_{nuc})Cln3_{nuc}Far1 + k_{off}Cln3Far1 - \eta(Cln3_{nuc}/Cln3Far1)Far1, \quad (18)$$

$$dCln3_{nuc}/dt = -(k_{on}/V_{nuc})Cln3_{nuc}Far1 + k_{off}Cln3Far1 + k_{cn}Cln3_{cyt} - k_{nc}Cln3_{nuc}, \quad (19)$$

$$Cln3_{cyt} = Cln3_{tot} - (Cln3_{nuc} + Cln3Far1), Cln3_{tot} = \theta P, \quad (20)$$

$$V_{nuc} = hV_{cell}, V_{cell} = P/H, \eta(x) = \eta^* x^{nF} / (1 + x^{nF}). \quad (21)$$

When 80% of the budded period has elapsed, a RESET function takes place and the $G1^*$ phase begins. RESET denotes the time instant when nuclear division (but not cell division) occurs: i.e. during the $G1^*$ phase each cell has two $G1$ nuclei and an undivided cytoplasm. At RESET the nuclear players show a discontinuity because they no more represent the whole (and unique) nuclear content and also because Far1 has been reset to a higher value. The RESET function includes instantaneous synthesis of Far1 and equal partition of Cln3Far1, Far1, Cln3_{nuc} in two nuclei whose volume is half of the original volume before RESET: $V_{nuc} = hV_{cell}/2$. Far1 degradation is inhibited ($\eta = 0$ in Eq. (18)), and Cln3 diffusion from the cytoplasm into the nucleus is strongly reduced (k_{cn} in Eq. (19) reduces of 5 orders of magnitudes during the $G1^*$).

Figure 4C–D shows the time course for the different molecular players throughout the whole cycle of an average size cell, growing in fast conditions (2% glucose, panel C) or in slow conditions (ethanol, panel D): when - very early after division - free nuclear Cln3 overcomes its inhibited form Cln3Far1 the first of the three consecutive Timers related to the cell cycle module is triggered. The time period spanning from the birth of the cell up to the aforementioned time instant is named T_{1a} . The kinetic parameters of the molecular trigger do not vary in different nutrient environments, except for the total amount of Far1, known to diminish in poor media [22], and for the parameters H , θ assumed to decrease in case of poor growth (see Table 2).

3.3 The Cell Cycle Module

In *S. cerevisiae* cell mass at division is unequally partitioned [19] between a larger, old parent cell (P) and a smaller, newly synthesized daughter cell (D). The degree of asymmetry of cell division in *S. cerevisiae* is modulated by nutrients: poor media – such as ethanol - yield a high level of asymmetry with large parent cells and very small daughter cells, whereas in rich media - such as glucose - parents and daughters at division are very close in size (reviewed in [15]). Since cells have to grow to a critical

Table 2. GroCy parameters.

Parameters	Meas. unit	glc 2%	Ethanol	Parameters	Meas. unit	glc 2%	Ethanol
ρ	rib/aa	$2.02e-5$	$1.18e-5$	$\tau_{2s}, s \geq I$	min	1500	3000
$P(0)$	aa	$2.76e10$	$0.8e10$	k_{on}	$(\text{molec/L})^{-1}/\text{min}$	$1.63e-15$	$1.63e-15$
$R(0)$	rib	$5.57e5$	$0.94e5$				
$FarI(0)$	molec	240	110	k_{off}	min^{-1}	25	25
$Cln3_{nuc}(0)$	molec	0	0	h	–	0.07	0.07
$Cln3FarI(0)$	molec	0	0	H	aa/L	$7.09e23$	$6.18e23$
$FarI_{reset}$	molec	240	110	n_F	–	10	10
K_1	min^{-1}	1	0.6	$\bar{\eta}$	min^{-1}	1	1
τ_1	min	4000	2000	Θ	molec/aa	$3.02e-8$	$2.66e-8$
K_2	aa/rib/min	380	316.66	k_{cn}	min^{-1}	1.5	1.5
K_2^1	aa/rib/min	342	285.46	k_{nc}	min^{-1}	0.6	0.6
K_2^2	aa/rib/min	178	149.34	$k_{cn,reset}$	min^{-1}	$5e-4$	$5e-4$
K_2^3	aa/rib/min	69	58.59	$T_{1b,min}$	min	1	8
K_2^4	aa/rib/min	51	43.47	W_O	min	1503	7045
K_2^5	aa/rib/min	42	35.91	W_I	min	62.1	306
$K_2^s, s > 5$	aa/rib/min	35	29.86	T_2	min	10	30
τ_2	min	3000	6000	T_B	min	85	136

cell size before entering S phase and budding, small daughter cells have a longer cycle time than the corresponding parent cells, most notably in poor media. This difference in cycle time between daughter and parent cells is due to differences in the G1 phase, whilst the budded period T_B has essentially the same length in both parents and daughter cells [18]. Differences in growth rate have marginal effects on the length of T_B and dramatic effects on the length of G1 (reviewed in [15]).

As explained, T_{1a} is the period from the birth of a cell till the time instant when free nuclear Cln3 exceeds its inhibited form Cln3Far1; the rest of the cycle is modeled by the sequence of three consecutive timers T_{1b} , T_2 , and T_B . The sum of the period T_{1a} + timer T_{1b} corresponds to timer T_1 in [18]. The G1 phase is given by $T_1 + T_2$. Timer T_B encompasses the budded phase.

The first timer T_{1b} starts when free Cln3 exceeds its inhibited form Cln3Far1. The length of T_{1b} is related to the size of the cell, so that larger cells have smaller T_{1b} periods, and vice versa. More in details, T_{1b} length is set according to the equation

$$T_{1b} = \max\{T_{1b,min}, W_0 - W_1 \ln(P_{T1a})\}, \quad (22)$$

with P_{T1a} denoting the size of the cell at the end of T_{1a} . Notice that P_{T1a} plays an active role in the setting of T_{1b} only for cells small enough, i.e. only when:

$$W_0 - W_1 \ln(P_{T1a}) > T_{1b,min} \rightarrow P_{T1a} < \exp\{(W_0 - T_{1b,min}) / W_1\}. \quad (23)$$

This happens, for instance, with most of daughter cells. In parent cells P_{T1a} , usually, greater than the upper bound in inequality (23), so that their T_{1b} length is fixed to $T_{1b, min}$ and does not depend on the size.

The length of timer T_2 does not depend on protein content [18]. At the end of timer T_2 , the critical protein size P is estimated. The budded period T_B , includes the S, G2, M and G1* phases. G1* has been modeled as the last 20% period of T_B phase. The end of the timer results in cell division. Like timer T_2 , timer T_B length does not depend on protein content, (no difference between daughters and parents). Part of the GroCy parameters are influenced by - and vary according to - the nutrient environment.

3.4 Genealogical Age Heterogeneity and Pedigree Simulations

When a yeast cell buds, a chitin ring, called bud scar, is formed at the bud isthmus remaining on the Parent after bud separation [15]. The genealogical age 'k' of a parent cell is the same as the number of bud scars 's', that can be visually counted, since each new bud starts at a new site. A cell without bud scars ($s = 0$) is a Daughter cell and it has not yet completed a cycle. We denote by "D_k" a Daughter of genealogical age 'k' (Fig. 5A). Each D_k ($k > 1$) is born from a P_{k-1} Parent. D₁ are born from any D_k.

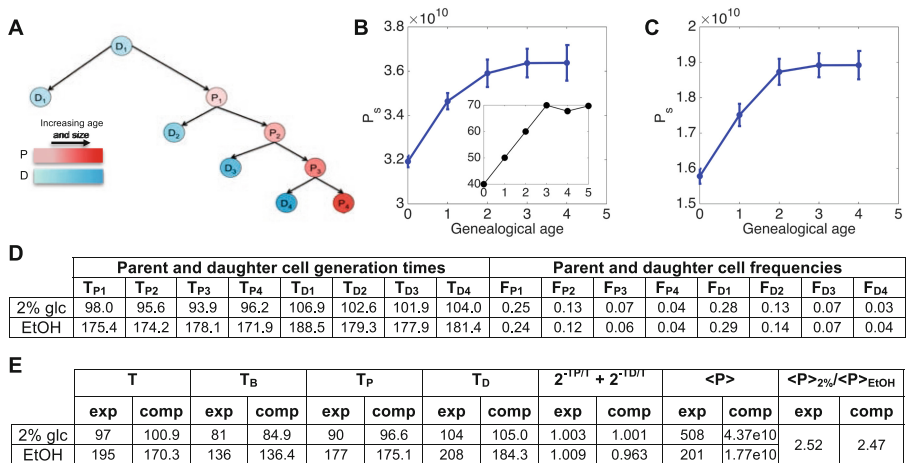


Fig. 5. A chain of cells P₁–P₄, D₁–D₄. **B, C:** Computed P_s (number of polymerized amino acid) for Parents of increasing genealogical age (until age 4), for fast (2% glucose, panel B) and slow (ethanol, panel C) growth conditions. Panel B reports the experimentally determined volume at bud initiation for Parents of increasing genealogical age (redrawn from [28]). **D:** Generation times, T_{Pk} , T_{Dk} , for Parents and Daughters of increasing genealogical ages obtained by simulating 20 different chains of cells P₁, ..., P₄ and D₁, ..., D₄. Frequencies of each sub-population (F_{Pk} , F_{Dk}) have been obtained using eqs. (A12, 13) in [30]. **E:** experimental and simulated values for relevant population parameters. T_D and T_P have been calculated from data in panel D, using eqs. (A4, 5) in [30].

Each Parent increases in size before starting to bud [28]. At division, it receives the mass it had at budding, while the mass synthesized during the budding phase goes to the newborn daughter. Hence, cell mass at budding (in Parents) and cell size at birth (in Daughters) increase with genealogical age. A reduced increase in parent cell size at budding with increasing genealogical age has been reported [15, 28] (see inset in Fig. 5B) and explained by mechanical stress of the cell wall, which increases with cell size [29]. Both K_2 and τ_2 in the growth module of GroCy (rate of protein synthesis and time constant of protein degradation respectively) decrease in value during the pre-budded period (G1 phase), according to the parent genealogical age, returning to their nominal values at the onset of the budding phase (end of Timer T_2), so that the parent cell P_k grows again with the steady-state exponential rate given by $\lambda = \rho K_2 - 1/\tau_2$. Daughter cells (of any genealogical age) are not affected by such a mechanical stress. The behavior of P_S qualitatively recapitulates experimental data (see Fig. 5B, C).

GroCy may be used to replicate small pedigree populations in different nutritional conditions, by suitably setting its parameters. We simulated 20 chains of cells $D_1, \dots, D_4, P_1, \dots, P_4$ (Fig. 5A), starting from 20 different initial cells, for two distinct growth conditions: 2% glucose and ethanol. The timers T_{1b}, T_2, T_B and the initial protein content $P(0)$ of each cell have been allowed to vary, with log-normal distribution, with a 5% CV over their average values. In order to estimate the average cell cycle length for both subpopulations of parents and daughters (T_P, T_D), we need to estimate the fractions of parents and daughters from the aforementioned “chain-cells” simulation. To this end we adopt the population modeling approximation described by Eq. (A1) given in [29] that provides the critical size of a parent P_k as a function of the critical size of a daughter and of the pair of parameters $a, Q < 1$. In exponential growth, the cell cycle length of parents and daughters of any genealogical age can be computed by means of Eqs. (A4, A5) of the same paper, where the parameter α denotes the exponential growth rate. Since these lengths are provided by the “chain-cells” simulation, we exploit the mentioned equations to infer the information on the population growth rate α and to estimate the values of (a, Q) that best fit these data. Parameters (a, Q) , as well as the growth rate α , are finally exploited to derive the fractions of cells ($F_{P1}, \dots, F_{P4}, F_{D1}, \dots, F_{D4}$) by way of the age distribution function [29]. The inferred structure population (Fig. 5D) allows to compute T_P and T_D , and the average protein content for the whole population, $\langle P \rangle$. Relative protein content, mass duplication times (T), T_P and T_D of yeasts growing on different media are very similar to experimental values. The relationship $2^{-TD/T} + 2^{-TP/T}$ - that links together T, T_P and T_D - yields a number very close to the theoretical value of 1 [15, 19], confirming that the simulated parameters capture the structure of yeasts growing on different carbon sources.

4 Conclusions

The growth activity combined to the other two main cellular activities of metabolism and cycle (MeGro and GroCy, respectively) define the modular building blocks constituting the coarse grain backbone of a modular, hierarchical and integrated *Metabolism, Growth and Cycle* model: iMeGroCy. The light green box in Fig. 6 reports a functional scheme highlighting the general procedure that allows to inter-connect

MeGro and GroCy as a function of the nutritional input (i.e., the glucose concentration). The current version of MeGro does not allow carbon sources other than glucose. MeGro responds to the external glucose $c_{glc,ex}$ and $Y_{EtOH/gluc}$ coming from experimental data in order to set the steady-state exponential growth rate λ and ribosome-over-protein ratio ρ as outputs of an optimization algorithm aiming at maximizing the growth rate. The MeGro outputs λ and ρ enter GroCy as inputs, allowing to set the ribosome and protein dynamics parameters, constituting the growth module. The exponential growth relationship $\lambda = \rho K_2 - 1/\tau_2$ is used to constrain the GroCy parameters K_2 and τ_2 to the MeGro outputs (λ , ρ). K_1 and τ_1 have been fixed accordingly to [18]. So, for cells growing on glucose-containing media, MeGro acts as a parameter generator for GroCy.

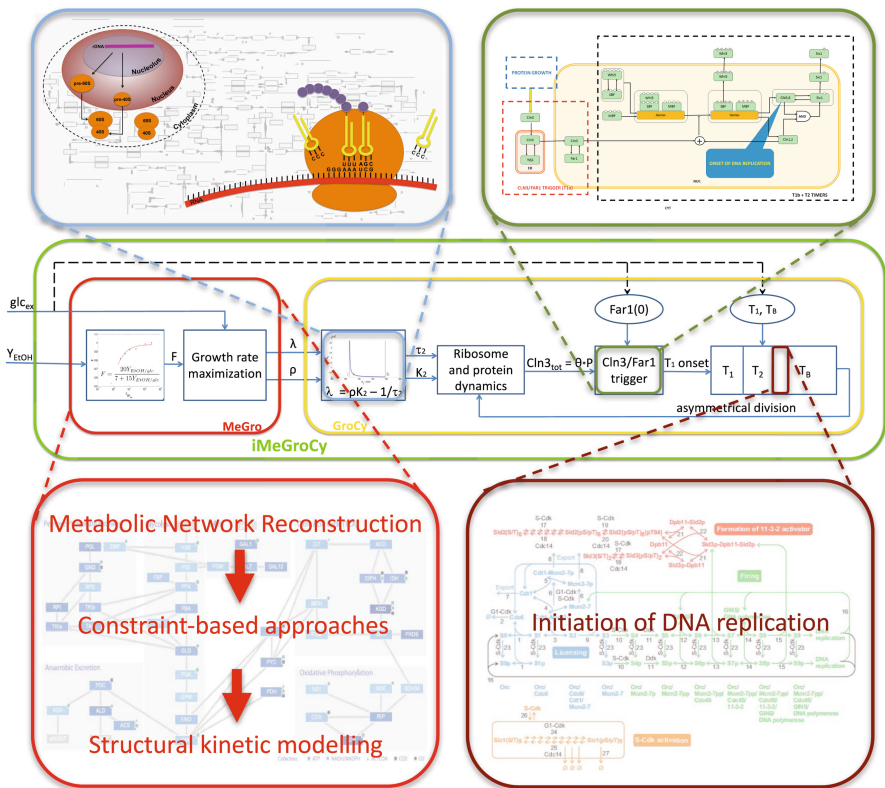


Fig. 6. Scheme of iMeGroCy (light green box). The scheme depicts the interconnection of the two main sub-blocks (MeGro, red block, and the GroCy, yellow block). The figure also shows iMeGroCy could host molecular blow-ups (plug-ins) of yeast functions. (Color figure online)

iMeGroCy differs from previous cell cycle models that either relied on defined molecular networks [31] - encompassing 27 components out of the much larger identified number [14, 32] or - when of low granularity [33] - did not show the same

degree of modularity offered by our approach. Other models concentrated on specific cell cycle phases and could be used in conjunction with iMeGroCy, whose modular and hierarchical nature allows it to act as a scaffold for the construction of a whole cell model for *S. cerevisiae* (Fig. 6). For instance, MeGro could be substituted by a genome-wide model [1], appropriately modified to include connections with cell growth and regulation by nutrients, the G₁ timers could be substituted by a recently described G₁/S module [20], entrance into S phase by a model of the onset of DNA synthesis [34], the budded phase by a wave of cyclins [35].

Adding the modules incrementally, the ability of iMeGroCy to fit experimental data could be monitored at any step. Top-down definition of the molecular modules would allow coherent expansion of iMeGroCy, favoring collaboration within the yeast community, since such an ambitious large-scale project will require a new type of collaborative effort [36].

References






1. Sánchez, B.J., Nielsen, J.: Genome scale models of yeast: towards standardized evaluation and consistent omic integration. *Integr. Biol. (Camb)* **7**, 846–858 (2015)
2. Botstein, D., Fink, G.R.: Yeast: an experimental organism for 21st Century biology. *Genetics* **189**, 695–704 (2011)
3. Karr, J.R., Sanghvi, J.C., Macklin, D.N., Gutschow, M.V., Jacobs, J.M., Bolival, B., Assad-Garcia, N., Glass, J.I., Covert, M.W.: A whole-cell computational model predicts phenotype from genotype. *Cell* **150**, 389–401 (2012)
4. Goffeau, A., Barrell, B.G., Bussey, H., Davis, R.W., Dujon, B., Feldmann, H., Galibert, F., Hoheisel, J.D., Jacq, C., Johnston, M., Louis, E.J., Mewes, H.W., Murakami, Y., Philippsen, P., Tettelin, H., Oliver, S.G.: Life with 6000 genes. *Science* **274**(546), 563–567 (1996)
5. Conrad, M., Schothorst, J., Kankipati, H.N., Van Zeebroeck, G., Rubio-Teixeira, M., Thevelein, J.M.: Nutrient sensing and signaling in the yeast *Saccharomyces cerevisiae*. *FEMS Microbiol. Rev.* **38**, 254–299 (2014)
6. Macklin, D.N., Ruggero, N.A., Covert, M.W.: The future of whole-cell modeling. *Curr. Opin. Biotechnol.* **28**, 111–115 (2014)
7. Molenaar, D., van Berlo, R., de Ridder, D., Teusink, B.: Shifts in growth strategies reflect tradeoffs in cellular economics. *Mol. Syst. Biol.* **5**, 323 (2009)
8. von der Haar, T.: A quantitative estimation of the global translational activity in logarithmically growing yeast cells. *BMC Syst. Biol.* **2**, 87 (2008)
9. Waldron, C., Jund, R., Lacroute, F.: The elongation rate of proteins of different molecular weight classes in yeast. *FEBS Lett.* **46**, 11–16 (1974)
10. Boehlke, K.W., Friesen, J.D.: Cellular content of ribonucleic acid and protein in *Saccharomyces cerevisiae* as a function of exponential growth rate: calculation of the apparent peptide chain elongation rate. *J. Bacteriol.* **121**, 429–433 (1975)
11. Waldron, C., Lacroute, F.: Effect of growth rate on the amounts of ribosomal and transfer ribonucleic acids in yeast. *J. Bacteriol.* **122**, 855–865 (1975)
12. Wortel, M.T., Bosdriesz, E., Teusink, B., Bruggeman, F.J.: Evolutionary pressures on microbial metabolic strategies in the chemostat. *Sci. Rep.* **6**, 29503 (2016)
13. Van Hoek, P., Van Dijken, J.P., Pronk, J.T.: Effect of specific growth rate on fermentative capacity of baker's yeast. *Appl. Environ. Microbiol.* **64**, 4226–4233 (1998)

14. Alberghina, L., Mavelli, G., Drovandi, G., Palumbo, P., Pessina, S., Tripodi, F., Coccetti, P., Vanoni, M.: Cell growth and cell cycle in *Saccharomyces cerevisiae*: basic regulatory design and protein-protein interaction network. *Biotechnol. Adv.* **30**, 52–72 (2012)
15. Porro, D., Vai, M., Vanoni, M., Alberghina, L., Hatzis, C.: Analysis and modeling of growing budding yeast populations at the single cell level. *Cytom Part J. Int. Soc. Anal. Cytol.* **75**, 114–120 (2009)
16. Porro, D., Brambilla, L., Alberghina, L.: Glucose metabolism and cell size in continuous cultures of *Saccharomyces cerevisiae*. *FEMS Microbiol. Lett.* **229**, 165–171 (2003)
17. Alberghina, L., Mariani, L., Martegani, E.: Cell cycle modelling. *Biosystems* **19**, 23–44 (1986)
18. Di Talia, S., Skotheim, J.M., Bean, J.M., Siggia, E.D., Cross, F.R.: The effects of molecular noise and size control on variability in the budding yeast cell cycle. *Nature* **448**, 947–951 (2007)
19. Hartwell, L.H., Unger, M.W.: Unequal division in *Saccharomyces cerevisiae* and its implications for the control of cell division. *J. Cell Biol.* **75**, 422–435 (1977)
20. Palumbo, P., Vanoni, M., Cusimano, V., Busti, S., Marano, F., Manes, C., Alberghina, L.: Whi5 phosphorylation embedded in the G₁/S network dynamically controls critical cell size and cell fate. *Nat. Commun.* **7**, ncomms11372 (2016)
21. Jorgensen, P., Edgington, N.P., Schneider, B.L., Rupes, I., Tyers, M., Futcher, B.: The size of the nucleus increases as yeast cells grow. *Mol. Biol. Cell* **18**, 3523–3532 (2007)
22. Alberghina, L., Rossi, R.L., Querin, L., Wanke, V., Vanoni, M.: A cell sizer network involving Cln3 and Far1 controls entrance into S phase in the mitotic cycle of budding yeast. *J. Cell Biol.* **167**, 433–443 (2004)
23. Fu, X., Ng, C., Feng, D., Liang, C.: Cdc48p is required for the cell cycle commitment point at Start via degradation of the G1-CDK inhibitor Far1p. *J. Cell Biol.* **163**, 21 (2003)
24. McKinney, J.D., Chang, F., Heintz, N., Cross, F.R.: Negative regulation of FAR1 at the Start of the yeast cell cycle. *Genes Dev.* **7**, 833–843 (1993)
25. Chang, F., Herskowitz, I.: Phosphorylation of FAR1 in response to alpha-factor: a possible requirement for cell-cycle arrest. *Mol. Biol. Cell* **3**, 445–450 (1992)
26. Peter, M., Gartner, A., Horecka, J., Ammerer, G., Herskowitz, I.: FAR1 links the signal transduction pathway to the cell cycle machinery in yeast. *Cell* **73**, 747–760 (1993)
27. Barberis, M., Klipp, E., Vanoni, M., Alberghina, L.: Cell size at S Phase Initiation: An Emergent Property of the G1/S Network. *PLoS Comput. Biol.* **3**, e64 (2007)
28. Johnston, G.C., Ehrhardt, C.W., Lorincz, A., Carter, B.L.: Regulation of cell size in the yeast *Saccharomyces cerevisiae*. *J. Bacteriol.* **137**, 1–5 (1979)
29. Alberghina, L., Vai, M., Vanoni, M.: Probing control mechanisms of cell cycle and ageing in budding yeast. *Curr. Genomics* **5**, 615–627 (2004)
30. Vanoni, M., Vai, M., Popolo, L., Alberghina, L.: Structural heterogeneity in populations of the budding yeast *Saccharomyces cerevisiae*. *J. Bacteriol.* **156**, 1282–1291 (1983)
31. Chen, K.C., Calzone, L., Csikasz-Nagy, A., Cross, F.R., Novak, B., Tyson, J.J.: Integrative analysis of cell cycle control in budding yeast. *Mol. Biol. Cell* **15**, 3841–3862 (2004)
32. Kaizu, K., Ghosh, S., Matsuoka, Y., Moriya, H., Shimizu-Yoshida, Y., Kitano, H.: A comprehensive molecular interaction map of the budding yeast cell cycle. *Mol. Syst. Biol.* **6**, 415 (2010)
33. Spiesser, T.W., Müller, C., Schreiber, G., Krantz, M., Klipp, E.: Size homeostasis can be intrinsic to growing cell populations and explained without size sensing or Signal. *FEBS J.* **279**, 4213–4230 (2012)

34. Brümmer, A., Salazar, C., Zinzalla, V., Alberghina, L., Höfer, T.: Mathematical modelling of DNA replication reveals a trade-off between coherence of origin activation and robustness against rereplication. *PLoS Comput. Biol.* **6**, e1000783 (2010)
35. Barberis, M., Linke, C., Adrover, M.Á., González-Novo, A., Lehrach, H., Krobitsch, S., Posas, F., Klipp, E.: Sic1 plays a role in timing and oscillatory behaviour of B-type cyclins. *Biotechnol. Adv.* **30**, 108–130 (2012)
36. Swierstra, T., Vermeulen, N., Braeckman, J., van Driel, R.: Rethinking the life sciences. To better serve society, biomedical research has to regain its trust and get organized to tackle larger projects. *EMBO Rep.* **14**, 310–314 (2013)



Estimating Effects of Extrinsic Noise on Model Genes and Circuits with Empirically Validated Kinetics

Samuel M. D. Oliveira^{1,2} , Mohamed N. M. Bahrudeen^{1,2} ,
Sofia Startceva^{1,2} , and Andre S. Ribeiro^{1,2}  

¹ Laboratory of Biosystem Dynamics, BioMediTech Institute, Tampere University of Technology, P.O Box 553, 33101 Tampere, Finland
andre.ribeiro@tut.fi

² Multi-scaled Biodata Analysis and Modelling Research Community, Tampere University of Technology, 33101 Tampere, Finland

Abstract. Recent studies of *Escherichia coli* transcription dynamics using time-lapse confocal microscopy and *in vivo* single-RNA detection confirmed that transcription initiation has two main rate-limiting steps. Here, we argue that this allows selective ‘tuning’ of the effects of extrinsic noise on a multi-scale level that ranges from individual genes to large-scale gene networks. First, using empirically validated stochastic models of transcription and translation, we show that the effects of RNA polymerase numbers’ cell-to-cell variability on the cell-to-cell diversity in RNA numbers decrease as the relative time-length of the open complex formation increases. Next, using a stochastic model of a 2-genes symmetric toggle switch, we show that the cell-to-cell diversity of the switching frequency due to cell-to-cell variability in RNA polymerase numbers also depends on the promoter kinetics. Finally, from the binarized protein numbers over time of 50-gene network models where genes interact by repression, we calculate the cell-to-cell variability of the mutual information and Lempel-Ziv complexity of the networks dynamics, and find that, while arising from the cell-to-cell variability in RNA polymerase numbers, these variability levels also depend on the promoter initiation kinetics. Given this, we hypothesize that *E. coli* may be capitalizing on the 2 rate-limiting steps’ nature of transcription initiation to tune the effects of extrinsic noise at the single gene, motifs, and large gene regulatory network levels.

Keywords: Transcription initiation · Extrinsic noise · Genetic circuits
Mutual information · Lempel-Ziv complexity

1 Introduction

When facing changing conditions, *Escherichia coli* cells can perform behavioral changes that can range from ‘smooth’ to ‘sharp’. This degree of change depends on the changes (how many and by how much) in the regulatory molecules of the transcriptional and translational machineries, such as RNA polymerase (RNAP) core enzymes, promoter sequence, σ factors, transcription factors, and ribosomes [1, 2].

Single-cell measurements have shown that, even in monoclonal bacterial populations, cells differ widely in component numbers [3, 4]. Consequently, the behavioral changes in response to, e.g., an environmental change, vary widely between individual cells. Such variability in cellular components numbers that causes cell-to-cell variability in the dynamics of cellular processes is usually termed “extrinsic noise”. Meanwhile, variability in the dynamics of a system that arises from the stochastic nature of its underlying processes (e.g. the stochastic nature of an event such as two molecules binding to one another) is usually termed ‘intrinsic noise’.

Because of the influence of these noise sources in cellular processes, the response of genes, in activity level, to global changes in regulatory molecules numbers is also highly diverse both in time (in a single cell, due to intrinsic noise) and across a cell population (due to intrinsic and extrinsic noise sources) [5]. In the case of σ factors’ direct positive regulation, this is believed to be due to a promoter-dependent selectivity for the σ factors [6], and/or the action of transcription factors [5]. Meanwhile, in the case of indirect negative regulation, it has recently been shown to be due to differences in the multi-step kinetics of transcription initiation of the promoters [7].

Following this finding, we recently have made use of stochastic modelling to explore the hypothesis that the dynamics of the rate-limiting steps in transcription initiation [8, 9] may influence individual genes’ degree of responsiveness to extrinsic noise [10, 11].

Here we investigate this phenomenon further, on a wide multi-scale perspective, namely, from individual genes to large-scale networks involving tens of genes. In particular, we study, at each level of complexity, the response to changing extrinsic noise levels as a function of the transcription kinetics of the component genes.

For this, we implement stochastic models of individual genes, genetic toggle switches, and 50-gene networks accounting for cell-to-cell diversity in RNAP numbers. Parameter values used in the models are obtained from recent microscopy measurements of single-cell RNAP, RNA, and protein numbers. Stochastic simulations [12, 13] of these models are performed to assess the extent to which the kinetics of initiation of the component promoters can be used to tune the level of the effects of the cell-to-cell variability in RNAP numbers on the dynamics of individual genes, genetic switches and 50-gene networks.

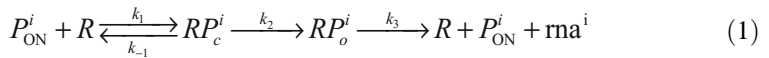
2 Methods

2.1 Models of Transcription, Translation, Genes Networks, and Source of Extrinsic Noise

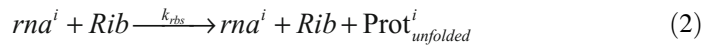
Our stochastic models of gene expression and genetic circuitry are based on multiple genome-wide studies of cell-to-cell variability in RNA numbers [14, 15], transcription dynamics of individual genes [9], translation kinetics at the single protein level [16–18], protein folding and activation kinetics [19], natural genetic switches [20, 21], and topology of large-scale circuits [22]. Importantly, the value set for each parameter associated to the core process of gene expression was obtained from empirical data (Table 1).

Assuming a N -genes network, the multi-step transcription process of an active promoter i , P_{ON}^i , is modeled by reactions (1), with $i = \{1, \dots, N\}$ [23]. The closed complex (RP_c^i) is formed once an RNAP (R) binds to a free promoter [24]. Subsequent steps follow to form the open complex (RP_o^i) [23, 24]. Finally, elongation starts [25], clearing the promoter. In the end, an RNA is produced and the RNAP is released. Elongation is not considered, due to its much shorter time-length when compared to initiation [9].

In (1), k_1 is the rate at which an RNAP (R) finds and binds to promoter P^i , k_{-1} is the rate of reversibility of the closed complex, k_2 is the rate of open complex formation, and k_3 is the rate of promoter escape (expected to be much higher than all other rates, and thus assumed to be ‘infinite’ [9]):



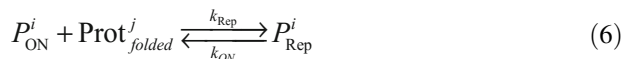
Reactions (2) and (3) model translation of the RNA and subsequent protein folding (which includes activation, for simplicity), respectively:



Reactions (4) and (5) model degradation and dilution due to cell division of RNA and proteins, respectively:



For simplicity, we assume that genes interact solely via repression mechanisms that block initiation [31]. This suffices to model several known small gene network motifs, such as, e.g., genetic switches. The repression mechanism is modeled by reactions (6), which account for the transition of the promoter to active/inactive (P_{ON}^i/P_{Rep}^i) due to the unbinding/binding of an active repressor protein ($Prot_{folded}^j$), produced by gene j :



Finally, we assume a mean cell lifetime (‘ div ’) of 1 h [9]. The dilution rate (Dil) of RNA and proteins due to cell division thus equals:

$$Dil = div^{-1} \times \log(2) \quad (7)$$

Taking into account the dilution due to cell division, along with the molecules' natural degradation rate (*Deg*), one has that the overall decay rate of, e.g., RNA molecules (k_d) along one line of a cell lineage will be:

$$k_d = Dil + Deg \quad (8)$$

The same formula is applied to proteins, using the appropriate rate constant (k_{dp}) for the degradation process. It is noted that, in agreement with [9], in the case of the model of single genes, we assume that the cell has a constant amount of active repressors, which, at points, force the promoter to go into the repressed state.

Given these models, assuming an 'active' promoter, we define τ_{prior} as the mean expected time for a successful closed complex formation, which depends on the speed and number of attempts to initiate an open complex formation (which, in turn, depends on the RNAP concentration). Meanwhile, the remaining time to produce an RNA, τ_{after} , includes the steps following commitment to open complex formation (e.g. isomerization [31]), and prior to transcription elongation. The mean time interval between consecutive RNA productions (Δt_{active}) of a fully active promoter is thus given by:

$$\Delta t_{\text{active}} = \tau_{\text{prior}} + \tau_{\text{after}} \quad (9)$$

Relevantly, in this model, τ_{after} does not depend on the RNAP intracellular concentrations. This is of significance in that, e.g., fluctuations in this concentration will only cause fluctuations in τ_{prior} and thus, will only cause 'partial' fluctuations in Δt_{active} , whose intensity will depend on the ratio $\tau_{\text{after}}/\Delta t_{\text{active}}$. Note also that we do not expect this formula to describe RNA production of genes in circuits, since the repression mechanism will cause significant changes to the RNA production kinetics prior to a successful closed complex formation.

The model above, as it is based on chemical reactions, and it is simulated in accordance with the stochastic simulation algorithm (SSA) [12], will result in systems whose dynamics are inherently stochastic due to two intrinsic noise sources, namely, the variability in the time moments that reactions occur and which reaction occurs next [12].

In addition to this, at the cell population level, the model possesses an extrinsic source of noise, which consists of a variability in the RNAP numbers of individual cells. This variability, as explained in the next section, is based on empirical data.

2.2 Cell-to-Cell Variability in RNA Polymerase Numbers

RNAP numbers in individual model cells are set based on measurements of RNAP fluorescence intensity in individual *E. coli* RL1314 cells with fluorescently tagged β' subunits [9]. In particular, we set the mean RNAP fluorescence in individual cells arbitrarily to 1 and obtain the fraction of cells with a given relative fluorescence level. The 2.5% cells with lowest and highest fluorescence intensity were discarded as outliers.

To obtain empirical values, we measured the cell-to-cell variability in RNAP numbers by calculating the squared coefficient of variation (CV^2) of the RNAP fluorescence intensity levels in individual cells. Next, to obtain the CV^2 of RNAP relative levels in individual cells, we fitted a normal distribution to the data (MATLAB package Statistics and Machine Learning Toolbox™). The CV^2 of the fit equaled 0.03, in agreement with [9]. To validate the fitting, we performed a Kolmogorov-Smirnov (KS) test between the empirical and best fit distributions, which showed that they cannot be statistically distinguished (p -value of 0.69, which, by being much higher than 0.01, clearly indicates that the null hypothesis that the two sets of data are from the same distribution cannot be rejected). Finally, we used this best fit distribution to set random RNAP numbers in each model cell, unless stated otherwise.

2.3 Detecting Switches in the Dynamics of the Toggle Switch and Switching Frequency Quantification

To detect ‘switches’ in the two protein numbers over time in toggle switches (where a switch is a change in which protein is more abundant), at each moment of the simulation, we calculate the difference between the numbers of the two proteins, denoted as ‘Prot1’ and ‘Prot2’. To not account short, transient switches, we use the following filter: if the absolute difference between Prot1 and Prot2 is smaller than 100 ($\sim 10\%$ of the mean protein numbers of an active gene in our models), we set the difference to 0. The number of switches during the time series is the number of times the difference between the two protein numbers changes from positive to negative or vice-versa. Given that n is the number of switches and Δt is the observation time, the switching frequency (F) is quantified as:

$$F = \frac{n + 1}{\Delta t} \quad (10)$$

2.4 Parameter Values and Simulations

Simulations are performed by SGNS [13], a simulator of chemical reaction systems based on the Delay Stochastic Simulation Algorithm [12, 32].

Each model cell ‘contains’ the systems of reactions (1)–(6). These reactions have the parameter values shown in Table 1 (unless stated otherwise). According to the model, e.g., increasing k_1 decreases the closed complex formation, while increasing k_2 shortens the open complex formation time. Here, we tune k_1 and k_2 so that the mean RNA production rate is kept constant, using the following formula [9]:

$$I(R) = \frac{(k_{ON} + k_{rep})(k_{-1} + k_2)}{Rk_1k_2k_{ON}} + \frac{1}{k_2} + \frac{1}{k_3} \quad (11)$$

where $I(R)$ is the mean interval between consecutive RNA productions in individual cells, assuming infinite cell lifetime. In (12), we first set all parameter values to the values shown in Table 1, to obtain the value of $I(R)$ in the control condition. Next, again

Table 1. Parameter values of the models (control condition). k_1 and k_{rbs} values are set assuming that the number of available RNAP and ribosomes equal 1 (and are never depleted).

Parameter	Value (s^{-1})	Reference
k_{ON}	0.01	[9]
k_{rep}	281	[9]
k_1	6469	[9]
k_{-1}	1	[9]
k_2	0.005	[9]
$k_{\text{d}_{\text{rna}}}$	0.0033	[14], Eq. (7)
k_{rbs}	0.637	[16–18]
k_{fold}	0.0024	[19]
k_{dp}	0.0019	[19], Eq. (7)

using this formula, we alter k_1 and k_2 , so that $I(R)$ is kept constant and equal to the $I(R)$ of the control condition. This allows changing the ratio $\tau_{\text{after}}/\Delta t$, which is given by:

$$\frac{\tau_{\text{after}}}{\Delta t} = 1 - \frac{(k_{\text{ON}} + k_{\text{OFF}})(k_{-1} + k_2)}{Rk_1k_2k_{\text{ON}}} \times I(R)^{-1} \quad (12)$$

The range of possible values of $\tau_{\text{after}}/\Delta t$ is set to be [0.05, 0.95], due to high diversity of empirical values of different promoters and promoters subject to different induction settings. These values, reported in [7], are here shown in Table 2:

Table 2. Empirical values of $\tau_{\text{after}}/\Delta t$ of various promoters under various induction levels.

Promoter and Induction	$\tau_{\text{after}}/\Delta t$	Reference
BAD (0.1% arabinose)	0.29	[7]
BAD (0.01% arabinose)	0.45	[7]
BAD (0.001% arabinose)	0.83	[7]
Lac- O_1O_3 (1 mM IPTG)	0.45	[7]
Lac- O_1O_3 (0.05 mM IPTG)	0.54	[7]
Lac- O_1O_3 (0.005 mM IPTG)	0.88	[7]
TetA (no inducers)	0.93	[7]
Lac- O_1 (1 mM IPTG)	0.95	[7]
Lac-ara1 (1 mM IPTG and 0.1% arabinose)	0.51	[7]

In the case individual genes and small motifs, we simulate models whose promoters differ in $\tau_{\text{after}}/\Delta t$ (by changing k_1 and k_2 while keeping Δt constant) by 0.1, from 0.05 to 0.95 (i.e. 10 conditions). Meanwhile, given that the RNAP cell-to-cell variability ($\text{CV}^2(\text{RNAP})$) is known to be equal to 0.03 in optimal growth conditions [9], and assuming that it is likely higher in sub-optimal conditions, we simulate models that differ in $\text{CV}^2(\text{RNAP})$ by 0.015, from 0 to 0.09 (i.e. 7 conditions). As such 70 different models are simulated. Specifications of the large-scale circuits (50-gene networks) simulated are described in the results section.

2.5 Mutual Information and Lempel-Ziv Complexity

In the case of large-scale gene networks (i.e. with 50 nodes and mean number of input connections of 2, see below), we calculate the Mutual Information (MI) and Lempel-Ziv (LZ) complexity of their dynamics. We consider such dynamics to correspond to the protein numbers of each gene over time (as in [26]). For that, we first define time windows (each window containing 10 consecutive time moments where protein numbers were collected) and calculate the mean protein numbers within that window, for each gene. Next, we binarize these numbers in accordance with a fixed threshold. If, in a given window, the mean protein numbers is smaller than 200, then the ‘binary protein value’ is set to zero. Else, it is set to 1. The threshold value of 200 for protein numbers was set based on our observation that, when the corresponding gene is repressed, its proteins usually tended to be smaller than 100, while when the gene is unrepressed, they tend to be larger than 300.

Based on this binarized data, to study the global propagation of information in the gene network, we make use of the average pairwise MI, which is a measure of the degree of correlation between the dynamics over time of all genes of the network. In particular, we defined MI as follows. Let S_a be a process that generates 0 with probability p_0 and 1 with probability p_1 . We define the entropy of S_a as:

$$H[s_a] = -p_0 \cdot \log_2 p_0 - p_1 \cdot \log_2 p_1 \quad (13)$$

Similarly, for a process S_{ab} that generates pairs xy with probabilities p_{xy} , where $x, y \in \{0, 1\}$, we define the joint entropy as:

$$H[s_{ab}] = -p_{00} \cdot \log_2 p_{00} - p_{01} \cdot \log_2 p_{01} - p_{10} \cdot \log_2 p_{10} - p_{11} \cdot \log_2 p_{11} \quad (14)$$

Finally, the MI of the pair of genes i and j is [26]:

$$MI_{ij} = H[s_i] + H[s_j] - H[s_{ij}] \quad (15)$$

Given this definition, MI_{ij} measures the extent to which information about node i at time t influences, directly or not, node j one time step later. From this, to quantify the efficiency of information propagation throughout the entire network, assuming N to be the number of nodes, we define the average pairwise MI of a network as:

$$MI = N^{-2} \cdot \sum_{i,j=1,\dots,N} MI_{ij} \quad (16)$$

In addition to the average pairwise MI, again using the windowed, binarized protein numbers data, we further calculate the Lempel-Ziv (LZ) complexity of each gene’s protein numbers over time (averaged over all genes) [27], as a means to quantify the degree of complexity of the signals that each gene of the network can generate.

In general, LZ measures a sequence’s complexity over a finite alphabet (here $\{0, 1\}$) by counting the number of new sub-strings (words) found, as the sequence is read (usually from left to right). For this, the algorithm used here [29] separates the sequence

into shortest words that haven't occurred yet, and the complexity equals the number of unique words, except for the last word, which may not be unique [27, 28]. Finally, assuming that n is the length of the time series of protein numbers from which the absolute LZ is calculated, we divide this absolute quantity by $\log_2(n)$ so as to scale it by the length, thus obtaining the scaled LZ for each gene. Then, we calculate the *average* scaled LZ of the network by summing the scaled LZ of each gene i and dividing by the total number of genes (N):

$$LZ = \frac{1}{N} \cdot \sum_{i=1, \dots, N} \left\{ LZ(i) \cdot \frac{\log_2(n)}{n} \right\} \quad (17)$$

3 Results and Conclusions

3.1 $\tau_{\text{After}}/\Delta t$ Tunes the Generation of Cell-to-Cell Variability in RNA Numbers from the Cell-to-Cell Variability in RNAP Numbers

We simulated the 70 models of individual genes described in the Methods section. Each model was simulated 100 times, each time for 2×10^4 s. From each simulation, we extracted the total number of produced RNA molecules during that time period.

As described in Methods, the models differ in such a way that the mean rate of RNA production should not differ. This was verified to be true. In all models, the total number of RNAs produced per cell equals ~ 20 , as expected (Table 1).

In Fig. 1, we show the CV^2 of the number of produced RNAs ($CV^2(\text{RNA})$) in individual cells in each model. We find that, as $\tau_{\text{after}}/\Delta t$ increases, the $CV^2(\text{RNA})$ decreases. Meanwhile, the $CV^2(\text{RNA})$ grows with increasing $CV^2(\text{RNAP})$. More importantly, visibly, both $\tau_{\text{after}}/\Delta t$ and $CV^2(\text{RNAP})$ need to be tuned in particular ways so that the $CV^2(\text{RNA})$ reaches a maximum and a minimum, which is not possible otherwise.

We conclude that, while the $CV^2(\text{RNAP})$ ‘propagates’ to the $CV^2(\text{RNA})$, as expected [33], the degree with which it does so strongly depends on the promoter initiation kinetics (specifically, it differs depending on the value of $\tau_{\text{after}}/\Delta t$).

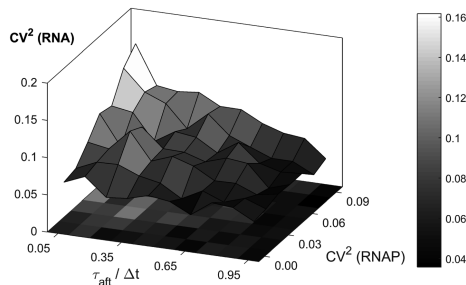


Fig. 1. CV^2 of number of produced RNAs in model cells versus $\tau_{\text{after}}/\Delta t$ and $CV^2(\text{RNAP})$.

3.2 $\tau_{\text{after}}/\Delta t$ Tunes the Influence of the Cell-to-Cell Variability in RNAP Numbers on the Cell-to-Cell Variability in Switching Frequency of a 2-Gene Toggle Switch

A 2-gene toggle switch consists of a genetic circuit of 2 genes that repress each other. Here, we model symmetric circuits, i.e., the 2 genes are identical. We simulated 70 models of toggle switches, differing in the dynamics of the component genes as described in Methods. Each model was simulated 100 times, each for 5×10^7 s, and protein numbers were assessed at a sampling interval of 10^4 s. In each simulation, we determined the moments when the protein numbers ‘switched’, as described in the Methods section, from which we obtained the mean and CV^2 of the switching frequency (F) for each model toggle switch. Results are shown in Fig. 2.

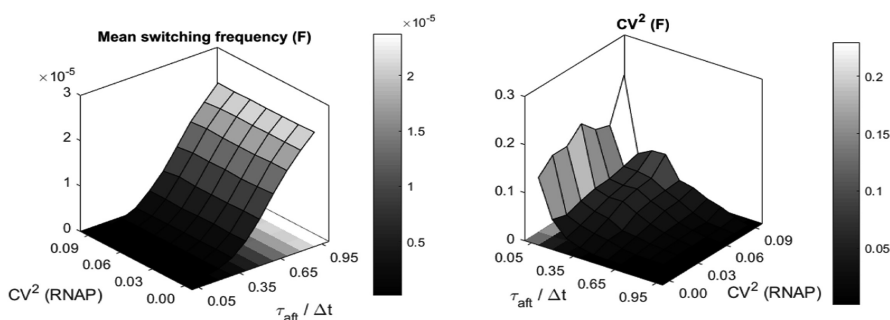


Fig. 2. (Left) Mean switching frequency (F) as a function of $\tau_{\text{after}}/\Delta t$ and $\text{CV}^2(\text{RNAP})$. (Right) Cell-to-cell diversity of the switching frequency ($\text{CV}^2(F)$) as a function of $\tau_{\text{after}}/\Delta t$ and $\text{CV}^2(\text{RNAP})$.

From Fig. 2(Left), decreasing $\tau_{\text{after}}/\Delta t$ decreases mean F , due to increased robustness of the ‘noisy attractors’ of the toggle switch [30], caused by a reduced probability of finding the ‘repressed’ promoter in a state that allows transcription to initiate. Meanwhile, changing $\text{CV}^2(\text{RNAP})$ does not affect the mean F , as this variable should not influence the mean behavior of the cell population, only the population’s behavior diversity.

From Fig. 2(Right), we find that as the $\text{CV}^2(\text{RNAP})$ increases, so does the $\text{CV}^2(F)$ (although mildly), provided that $\tau_{\text{after}}/\Delta t$ is smaller than ~ 0.7 – 0.8 . This is because, the smaller is $\tau_{\text{after}}/\Delta t$, the weaker is the filtering of the extrinsic noise. Overall, the $\text{CV}^2(\text{RNAP})$ affects the $\text{CV}^2(F)$ weakly since, first, the $\text{CV}^2(\text{RNAP})$ only affects the cell-to-cell variability of the mean $\tau_{\text{prior}}/\Delta t$ and, thus, if this step is short-length, the effects on the variability in RNA production kinetics will be weak. Second, for relatively small values of $\tau_{\text{prior}}/\Delta t$ compared to $\tau_{\text{after}}/\Delta t$, the switch becomes less stable [11], causing the effects of increasing $\text{CV}^2(\text{RNAP})$ to become negligible.

Meanwhile, $\tau_{\text{after}}/\Delta t$ has a strong influence on $\text{CV}^2(F)$ since, the smaller is its value, the larger is $\text{CV}^2(F)$. This is explained by the fact that $\tau_{\text{prior}}/\Delta t$ will be larger for small values of $\tau_{\text{after}}/\Delta t$, and thus, the transcription kinetics will be more influenced by the

noise from events such as RNAP binding, repressors binding, etc., causing the switch to change have a more noisy dynamics.

3.3 Large-Scale Circuits. Information Generation and Propagation

We simulate networks with 50 nodes and mean number of input connections (i.e. mean connectivity) of 2. The network topologies are generated using the ‘Erdős Random 2’ algorithm proposed in [34] that allow producing Erdős random graphs [35].

In these networks, each node is a gene whose expression dynamics and interactions (i.e. connections) are defined according to reactions (1–6). Consequently, all interactions between genes consist of repression mechanisms, as those used to model the genetic switches above. Given this modelling strategy, at any given moment, a gene is expected to be either actively expressing (in the absence of its repressor proteins, which can be more than 1) or to be repressed. We hypothesize that the degree of repression should depend on the number of genes that can directly repress a given gene, on the relative time length during which the repressing genes are active, and, importantly, as seen above, on the value of $\tau_{\text{after}}/\Delta t$ of the repressed gene.

To study how changing $\tau_{\text{after}}/\Delta t$ and $\text{CV}^2(\text{RNAP})$ affects the networks ability to generate and propagate information, for each set of values of $\tau_{\text{after}}/\Delta t$ and $\text{CV}^2(\text{RNAP})$, we generate 10 topologies. Then, we simulated each topology 10 times, with individual simulations differing in RNAP numbers as above.

Each simulation lasts 10^6 s, and the numbers of each protein were collected each 10^3 s. Each such set of collected protein numbers constitutes a network ‘time moment’. From these, network ‘states’ over time are obtained as follows. First, time windows with a length of 10 time moments are defined (the first window is not considered since the network is initialized without proteins). Next, the protein numbers in each state are obtained by averaging these numbers from all 10 time moments composing the window. Finally, for each protein, we binarize its mean numbers of each time window using a fixed threshold (see Methods).

Thus, from each simulation, we obtained 100 consecutive ‘binary states’ of the given network. From this data, we can then calculate the average pairwise MI and the average scaled LZ, so as to measure, respectively, the degrees of information propagation and generation of the network during that time period. Finally, for each condition, we obtained the averages of these two quantities for all networks.

Six models of gene expression differing in $\tau_{\text{after}}/\Delta t$ and $\text{CV}^2(\text{RNAP})$ were considered (Table 3). Note that, as described in Methods, all genes of a given network share the same value of $\tau_{\text{after}}/\Delta t$ and all networks of a given model share the same value of $\text{CV}^2(\text{RNAP})$. The values for these two parameters were chosen so as to test whether they can affect the mean and variability of the information generation and propagation capabilities of the networks. Results are shown in Table 3.

From Table 3, we find that the value of $\tau_{\text{after}}/\Delta t$ of the component genes affects the values of $\mu(\text{MI})$, $\mu(\text{LZ})$, $\text{CV}^2(\text{MI})$, and $\text{CV}^2(\text{LZ})$. This is expected (from reactions (1) and (6)), since this parameter affects the degree to which a gene’s activity is affected not only by variability in RNAP numbers but also by its repressor genes’ activity levels. Meanwhile, the $\text{CV}^2(\text{RNAP})$ affects the $\text{CV}^2(\text{MI})$, provided small values of $\tau_{\text{after}}/\Delta t$, as expected given the results for the toggle switch model.

Table 3. Pairwise mutual information (MI) and scaled Lempel-Ziv complexity (LZ) mean (μ) and CV^2 as a function of $\tau_{\text{after}}/\Delta t$ of the promoters and the $CV^2(\text{RNAP})$ of the cell populations.

Condition	$\tau_{\text{after}}/\Delta t$	$CV^2(\text{RNAP})$	$\mu(\text{MI})$	$\mu(\text{LZ})$	$CV^2(\text{MI})$	$CV^2(\text{LZ})$
1a	0.1	0.01	0.004	0.18	3.4	0.0016
1b	0.1	0.08	0.005	0.18	1.6	0.0018
2a	0.5	0.01	0.008	0.48	0.002	0.0007
2b	0.5	0.08	0.008	0.48	0.002	0.0009
3a	0.9	0.01	0.0007	0.17	0.05	0.0016
3b	0.9	0.08	0.0007	0.17	0.04	0.0016

4 Discussion

We performed simulations of stochastic models of single gene, 2-gene toggle switches, and large-scale (50 genes) genetic circuits, all of which include the multi-step process of transcription, whose parameter values have been obtained from empirical data extracted from *in vivo*, single-cell measurements on *E. coli* cells.

Overall, we find that the relative time that the gene spends in the rate-limiting steps *after* initiation of the open complex formation, here quantified by the ratio $\tau_{\text{after}}/\Delta t$, significantly affects the degree to which individual genes and circuits (small- and large-scale) are affected by extrinsic noise.

It is of interest that this property of the promoter initiation kinetics has a clear multi-scale effect, ranging from effects on RNA numbers of individual genes over time, to effects on the dynamics of small network motifs, to effects on the capacity of large networks to produce and propagate information. The above suggests that this feature of the kinetics of transcription may be used as a ‘master regulator’ of the functioning of the genetic circuits in *E. coli*, perhaps as influent as the global and local topological structures formed by promoter-protein, protein-protein, and RNA-RNA interactions.

Importantly, the kinetics of transcription initiation of each gene in the network is both sequence dependent as well as subject to regulation, both by transcription factors as well as by global regulatory molecules, such as σ factors. As such, this mechanism is both, respectively, evolvable as well as adaptive at the single gene level. We hypothesize that, given this, the 2 rate-limiting step nature of the transcription process may confer *E. coli* rapid evolvability as well as plasticity in fluctuating environments.

In the future, we plan to perform a wide range of experiments to validate our findings, as well as to make use of additional simulation and more detailed models to further explore how the dynamics of transcription of individual genes may act as a regulator of the degree of influence of extrinsic noise on genetic networks.

Acknowledgements. Work supported by Academy of Finland (295027 ASR), Academy of Finland Key Project Funding (305342 ASR), Jane and Aatos Erkko Foundation (610536 ASR), Finnish Academy of Science and Letters (SO), and Tampere University of Technology President’s Graduate Program (SS). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

1. Jishage, M., Iwata, A., Ueda, S., Ishihama, A.: Regulation of RNA polymerase sigma sub-unit synthesis in *Escherichia coli*: intracellular levels of four species of sigma subunit under various growth conditions. *J. Bacteriol.* **178**, 5447–5451 (1996)
2. Rahman, M., Hasan, M.R., Oba, T., Shimizu, K.: Effect of rpoS gene knockout on the metabolism of *Escherichia coli* during exponential growth phase and early stationary phase based on gene expressions, enzyme activities and intracellular metabolite concentrations. *Biotechnol. Bioeng.* **94**, 585–595 (2006)
3. Megerle, J.A., Fritz, G., Gerland, U., Jung, K., Rädler, J.O.: Timing and dynamics of single cell gene expression in the arabinose utilization system. *Biophys. J.* **95**, 2103–2115 (2008)
4. Jones, D.L., Brewster, R.C., Phillips, R.: Promoter architecture dictates cell-to-cell variability in gene expression. *Science* **346**, 1533–1537 (2014)
5. Farewell, A., Kvint, K., Nyström, T.: Negative regulation by RpoS: a case of sigma factor competition. *Mol. Microbiol.* **29**, 1039–1051 (1998)
6. Hengge-Aronis, R.: Recent insights into the general stress response regulatory network in *Escherichia coli*. *J. Mol. Microbiol. Biotechnol.* **4**, 341–346 (2002)
7. Kandavalli, V.K., Tran, H., Ribeiro, A.S.: Effects of σ factor competition on the *in vivo* kinetics of transcription initiation in *E. coli*. *BBA Gene Regul. Mech.* **1859**, 1281–1288 (2016)
8. McClure, W.R.: Rate-limiting steps in RNA chain initiation. *Proc. Natl. Acad. Sci. USA* **77**, 5634–5648 (1980)
9. Lloyd-Price, J., Startceva, S., Kandavalli, V., Chandraseelan, J., Goncalves, N., Oliveira, S. M.D., Häkkinen, A., Ribeiro, A.S.: Dissecting the stochastic transcription initiation process in live *Escherichia coli*. *DNA Res.* **23**(3), 203–214 (2016)
10. Bahrudeen, M.N.M., Startceva, S., Ribeiro, A.S.: Effects of extrinsic noise are promoter kinetics dependent. In: *The 9th International Conference on Bioinformatics and Biomedical Technology on Proceedings, ICBBT 2017, Lisbon, Portugal*, pp. 44–47 (2017)
11. Bahrudeen, M.N.M., Startceva, S., Ribeiro, A.S.: Tuning extrinsic noise effects on a small genetic circuit. In: *The European Conference on Artificial Life on Proceedings, ECAL 2017, Lyon, France*, vol. 14, pp. 454–459 (2017)
12. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **81**(25), 2340–2361 (1977)
13. Lloyd-Price, J., Gupta, A., Ribeiro, A.S.: SGNS2: a compartmentalized stochastic chemical kinetics simulator for dynamic cell populations. *Bioinformatics* **28**, 3004–3005 (2012)
14. Bernstein, J.A., Khodursky, A.B., Pei-Hsun, L., Lin-Chao, S., Cohen, S.N.: Global analysis of mRNA decay and abundance in *Escherichia coli* at single-gene resolution using two-color fluorescent DNA microarrays. *Proc. Natl. Acad. Sci. USA* **99**, 9697–9702 (2002)
15. Taniguchi, Y., Choi, P.J., Li, G.-W., et al.: Quantifying *E. coli* proteome and transcriptome with single-molecule sensitivity in single cells. *Science* **329**, 533–538 (2010)
16. Mitarai, N., Sneppen, K., Pedersen, S.: Ribosome collisions and translation efficiency: optimization by codon usage and mRNA destabilization. *J. Mol. Biol.* **382**, 236–245 (2008)
17. Bremer, H., Dennis, P.P.: Modulation of chemical composition and other parameters of the cell by growth rate. In: Neidhardt, F.C. (ed.) *Escherichia Coli and Salmonella*, 2nd edn, pp. 1553–1569. ASM Press, Washington, DC (1996)
18. Kennel, D., Riezman, H.: Transcription and translation initiation frequencies of the *Escherichia coli* lac operon. *J. Mol. Biol.* **114**(1), 1–21 (1977)
19. Cormack, B.P., Valdivia, R.H., Falkow, S.: FACS-optimized mutants of the green fluorescent protein (GFP). *Gene* **173**(1), 33–38 (1996)

20. Neubauer, Z., Calef, E.: Immunity phase-shift in defective lysogens: non-mutational hereditary change of early regulation of λ Prophage. *J. Mol. Biol.* **51**, 1–13 (1970)
21. Arkin, A., Ross, J., McAdams, H.: Stochastic kinetic analysis of developmental pathway bifurcation in phage λ -infected *Escherichia coli* cells. *Genetics* **149**, 1633–1648 (1998)
22. Fu, Y., Jarboe, L.R., Dickerson, J.A.: Reconstructing genome-wide regulatory network of *E. coli* using transcriptome data and predicted transcription factor activities. *BMC Bioinform.* **12**(233) (2011). <https://doi.org/10.1186/1471-2105-12-233>
23. Saecker, R.M., Record, M.T., Dehaseth, P.L.: Mechanism of bacterial transcription initiation: RNA polymerase - promoter binding, isomerization to initiation-competent open complexes, and initiation of RNA synthesis. *J. Mol. Biol.* **412**, 754–771 (2011)
24. Chamberlin, M.: The selectivity of transcription. *Annu. Rev. Biochem.* **43**, 721–775 (1974)
25. deHaseth, P.L., Zupancic, M.L., Record, M.T.: RNA polymerase promoter interactions: the comings and goings of RNA polymerase. *J. Bacteriol.* **180**, 3019–3025 (1998)
26. Ribeiro, A.S., Kauffman, S.A., Lloyd-Price, J., Samuelsson, B., Socolar, J.E.S.: Mutual information in random Boolean models of regulatory networks. *Phys. Rev. E* **77**, 011901 (2008)
27. Lempel, A., Ziv, J.: On the Complexity of Finite Sequences. *IEEE Trans. Inform. Theory* **22**, 75–81 (1976)
28. Shmulevich, I., Kauffman, S.A., Aldana, M.: Eukaryotic cells are dynamically ordered or critical but not chaotic. *Proc. Natl. Acad. Sci. USA* **102**(38), 13439–13444 (2005)
29. Borowska, M., Oczeretko, E., Mazurek, A., Kitlas, A., Kuć, P.: Application of the Lempel-Ziv complexity measure to the analysis of biosignals and medical images. In: *Annual Proceedings of Medical Science*, vol. 50, Suppl. 2 (2005)
30. Ribeiro, A.S., Kauffman, S.A.: Noisy attractors and ergodic sets in models of gene regulatory networks. *J. Theor. Biol.* **247**(4), 743–755 (2007)
31. McClure, W.R.: Mechanism and control of transcription initiation in prokaryotes. *Annu. Rev. Biochem.* **54**, 171–204 (1985)
32. Roussel, M.R., Zhu, R.: Validation of an algorithm for delay stochastic simulation of transcription and translation in prokaryotic gene expression. *Phys. Biol.* **3**, 274–284 (2006)
33. Elowitz, M.B., Levine, A.J., Siggia, E.D., Swain, P.S.: Stochastic gene expression in a single cell. *Science* **297**, 1183–1186 (2002)
34. Airoidi, E.M., Carley, K.M.: Sampling algorithms for pure network topologies: a study on the stability and the separability of metric embeddings. *ACM SIGKDD Explor. Newsl.* **7**(2), 13–22 (2005)
35. Bollobas, B.: *Random Graphs*, 2nd edn. Academic Press, New York (2001)

Economy and Society



Calibrating Dynamic Factor Models with Genetic Algorithms

Fabio Della Marra^{1,2} 

¹ European Centre for Living Technology, Ca' Minich, S. Marco 2940,
30124 Venice, Italy

fabio.dellamarra@unive.it

² Department of Environmental Sciences, Informatics and Statistics,
Ca' Foscari University of Venice, Cannaregio 873, 30121 Venice, Italy

Abstract. In this work, we address the problem of calibrating dynamic factor models for macroeconomic forecasting. The variables upon which the forecasts are computed are the logarithm of the Industrial Production (IP) and the yearly change of the logarithm of the Consumer Price Index (CPI). Our purpose is to provide a contribution to the model identification by proposing a new kind of calibration of static and dynamic factor models. The innovative part of our work consists of building a genetic algorithm for calibrating three dynamic factor models. We first analyse a dataset of 176 EU macroeconomic and financial time series and then we conduct the same study on a dataset of 115 US macroeconomic and financial time series. In both studies, the employment of genetic algorithm in the calibration procedure produces very good results and more significant than those achieved in similar studies, such as [1, 2].

Keywords: Macroeconomic time-series forecasting
Genetic algorithms · Dynamic factor models

1 Introduction

In this work, we propose a novel approach to the calibration of three selected large-dimensional dynamic factor models for macroeconomic forecasting by means of a genetic algorithm. Some insights about the three selected dynamic factor models are reported below:

- (i) *Stock and Watson (SW) model.* This time-domain method was introduced in [3, 4]. The factors are estimated by computing static principal components of the variables in the dataset. Let y_{it} be the variable of the dataset to be forecasted at time t , its h -step-ahead prediction equation (also called *Diffusion Forecast Index*) is obtained by regressing y_{it+h} on the factors and on y_{it} itself. Lags of the factors and of y_{it} may be added.
- (ii) *Forni, Hallin, Lippi and Reichlin (FHLR) model.* This frequency-domain method was proposed in [5, 6] and requires the computation of two steps. In a

first step, the common component χ_t , the idiosyncratic component ξ_t and their covariances are estimated using a frequency-domain method introduced in [5] named *Dynamic Principal Component*. In the second step, the factors are estimated by computing Generalized Principal Components.

(iii) *Forni, Hallin, Lippi and Zaffaroni (FHLZ) model*. This frequency-domain method was proposed in [7, 8]. Here, the underlying assumption in (i) and (ii) that the common components span a finite-dimensional space as n tends to infinity is relaxed.

There exists some literature comparing the forecasting performances of SW and FHLR, but universal consensus still does not seem to have been reached. Theoretically, time-domain methods (as FHLR and FHLZ) consider only relations among the variables at the same time, whereas frequency-domain methods (as SW) exploit leaded and lagged relations among the variables. However time-domain methods require less parameters to be calibrated. Hence they are more robust to misspecification than frequency-domain methods. Instead, a systematic comparison of the forecasting performances of SW, FHLR and FHLZ can be found only in [1, 2]. [2] conducted a forecasting exercise on a US macroeconomic dataset, taking an autoregressive process of order 4 as a benchmark. They showed that FHLZ outperforms SW, FHLR and the benchmark both for the Industrial Production and the CPI during the Great Moderation (1982–2007). In the Great Recession (2007–2012), the forecasting performances of the Industrial Production change dramatically: all factor models are outperformed by the benchmark. SW and FHLR outperform FHLZ. Hence, Forni et al. concluded that, due to its more dynamical structure, FHLZ tends to be the best performing method in “stationary periods”, but it loses ground during regime changes. [1] conducted a forecasting exercise on an EU macroeconomic dataset. The global settings of his exercise are basically the same as in [2], but also the length of the rolling window is suboptimally selected during the calibration process. He found that, on the proper sample, FHLZ is the most performing for the CPI. However, mixed evidences appear over the proper sample for the Industrial Production. Since each model is characterized by several parameters to estimate, an exhaustive exploration of the parameter space would be computationally infeasible. In order to give a partial solution to this issue, in [1] and in [2] the calibration procedure is carried out in a *naïf* fashion, i.e. an initial configuration for each parameter is randomly selected and then, for each parameter at a time, a predetermined range of values is tested while keeping the other parameters fixed. As all the parameters have been tested, the configuration of the parameters with the lowest mean-squared forecast error (MSFE) is selected. The drawback of this procedure is that the final configuration selected may depend on the order on which the parameters have been processed in the calibration process. The novelty introduced in this paper is the employment of a genetic algorithm to explore the parameter space. In fact, the genetic algorithm allows us to select a suboptimal configuration of the parameters without imposing any order on the parameters to be estimated. In this work, we also compare the macroeconomic forecasting performance of the three selected dynamic

factor models on two datasets. The former (an EU macroeconomic and financial dataset) is the same employed in [1]. Instead, the latter (an US macroeconomic and financial dataset) is the same employed in [2]. The paper is structured as follows. In Sect. 2, the calibration process of the models with a genetic algorithm is described. In Sect. 3, the results achieved on the EU dataset are discussed and, the same analysis is developed in Sect. 4 for the US dataset. In Sect. 5, some concluding remarks are presented.

2 The Calibration Process with a Genetic Algorithm

Both datasets contain real variables (import/export price indexes, employment, Industrial Production) and nominal variables (money aggregates, consumer price indexes, wages), asset prices (stock prices and exchange rates) and surveys. To achieve stationarity, several series are deseasonalized and transformed. No treatment for outliers is applied. In addition to SW, FHLR, FHLZ, the forecasts of an autoregressive process (AR) are computed. The order p of the AR process is determined in the calibration process. As in [1,4], to assess the forecasting performances, the variables which are taken into account are the level of the logarithm of the Industrial Production (IP) and the yearly change of the logarithm of the Consumer Price Index (CPI). Forecasts are computed h -months ahead, with $h \in \{1, 3, 6, 12, 24\}$. For each methods, we employ a rolling-window scheme $[t-l, t]$, whose size l is determined in the calibration sample.

As to the calibration process, the observations of the EU dataset ranging from February 1986 to December 2000 will be used to calibrate the methods SW, FHLR, FHLZ and the benchmark. For this reason, we will refer to this range of the EU dataset as the *calibration sample*. Instead, the calibration sample in the US dataset will range from March 1960 to December 1984. At each epoque, the population of the genetic algorithm is a subset of the strings containing all the possible configurations of the parameters. We set the MSFE as the objective function to be minimised by the genetic algorithm. For each method, we iterate the genetic algorithm ten times on the calibration sample of the two datasets. The fitness of each individual is stored in a data structure. Eventually, for each method we select as the most performing configuration the one endowed with the lowest MSFE. More precisely, we select the configuration with the lowest objective function value that has been assessed during each of the ten runs of the genetic algorithms, independently from the final solutions obtained at each run. The parameters of each run of the genetic algorithm are the following:

- (i) *Population size of the genetic algorithm at each generation = 100;*
- (ii) *Crossover fraction = 0.6;*
- (iii) *Number of individuals who passes to the next generation = 25;*
- (iv) *Mutation = Gaussian model (adds a random number chosen from a Gaussian distribution, to each entry of the parent vector).*

The stopping criteria of each run of the genetic algorithm are the following:

- (i) *Maximum number of generations = 1000;*
- (ii) *Maximum number of generations in which the difference between the average MSFE is less than the threshold $10^{-7} = 5$;*
- (iii) *MSFE of an individual in the last generation tending to zero.*

The same procedure is described in [9] and in [10], but the main purpose of these articles is to achieve suboptimal variable selection in a regressive setting.

3 Results on the EU Dataset

In this chapter, we will use the same notation as in [1].

3.1 Calibration of SW Model

In SW, the following parameters must be calibrated:

- (i) *The number of static factors r :* ranging from 1 to 10. Also, a comparison with Bai & Ng criterium (BN) with maximum 12 factors has been made.
- (ii) *The degree α of $\mathbf{a}(L)$:* ranging from 1 to 10.
- (iii) *The degree β of $b(L)$:* ranging from 0 to 10.
- (iv) *The size l of the rolling window:* ranging from 5 to 12 years.

After the ten runs of the genetic algorithms in the calibration process, the individual granted with the minimum objective function value for the IP is the following:

$$(r, \alpha, \beta, l) = (5, 1, 0, 11). \quad (3.1)$$

Instead, the individual granted with the minimum objective function value for the CPI is the following:

$$(r, \alpha, \beta, l) = (1, 0, 1, 7). \quad (3.2)$$

3.2 Calibration of FHLR Model

In FHLR, the following parameters must be calibrated:

- (i) *The number of static factors r :* ranging from 1 to 10. Also, a comparison with Bai & Ng criterium (BN) with maximum 12 factors has been carried out.
- (ii) *The number of dynamic factors q :* ranging from 0 to 10. Also, a comparison with Hallin-Liska criterium (HL) with maximum 12 factors has been carried out.
- (iii) *The type of kernel k :* ranging in the set {Triangular, Rectangular, Parzen, Gaussian, Exponential, Cosine, Tukey, Hann}.

- (iv) *The lag window d for spectral density estimation:* ranging in the set $\{25, 35, 40\}$.
- (v) *The size l of the rolling window:* ranging from 5 to 12 years.

After the ten runs of the genetic algorithms in the calibration process, the individual granted with the minimum objective function value for the IP is the following:

$$(r, q, k, d, l) = (10, 3, \textit{Cosine}, 35, 11). \tag{3.3}$$

Instead, the individual granted with the minimum objective function value for the CPI is the following:

$$(r, q, k, d, l) = (8, 4, \textit{Hann}, 25, 7). \tag{3.4}$$

3.3 Calibration of FHLZ Model

In FHLZ, the following parameters must be calibrated:

- (i) *The number of dynamic factors q :* ranging from 1 to 5. Also, a comparison with Hallin-Liska criterium has been carried out.
- (ii) *The type of kernel k :* ranging in the set $\{\textit{Triangular}, \textit{Rectangular}, \textit{Parzen}, \textit{Gaussian}, \textit{Exponential}, \textit{Cosine}, \textit{Tukey}, \textit{Hann}\}$.
- (iii) *The lag window d for spectral density estimation:* ranging in the set $\{25, 35, 40\}$.
- (iv) *The maximum lag ml for the matrix $\mathbf{A}^k(L)$:* ranging from 1 to 5.
- (v) *The size l of the rolling window:* ranging from 5 to 12 years.

After the ten runs of the genetic algorithms in the calibration process, the individual granted with the minimum objective function value for the IP is the following:

$$(q, k, d, ml, l) = (4, \textit{Parzen}, 25, 4, 11). \tag{3.5}$$

Instead, the individual granted with the minimum objective function value for the CPI is the following:

$$(q, k, d, ml, l) = (2, \textit{Parzen}, 35, 1, 7). \tag{3.6}$$

3.4 Calibration of the Benchmark

To calibrate the benchmark $AR(p)$, the only parameter that needs to be fixed is the order p . In our exercise, we let p range from 1 to 13. By selecting the values of the parameter p which guarantee the lowest mean RMSFE, the chosen configuration for the IP is the following:

$$p = 2. \tag{3.7}$$

Instead, the chosen configuration for the CPI is the following:

$$p = 1. \tag{3.8}$$

3.5 Empirical Proof of the Convergence of the Runs of the Genetic Algorithm

To give an empirical proof of the convergence of the genetic algorithm, in Fig. 1 the boxplots of the results of the ten runs of each selected dynamic factor models for the IP (on the left) and for the CPI (on the right) are reported.

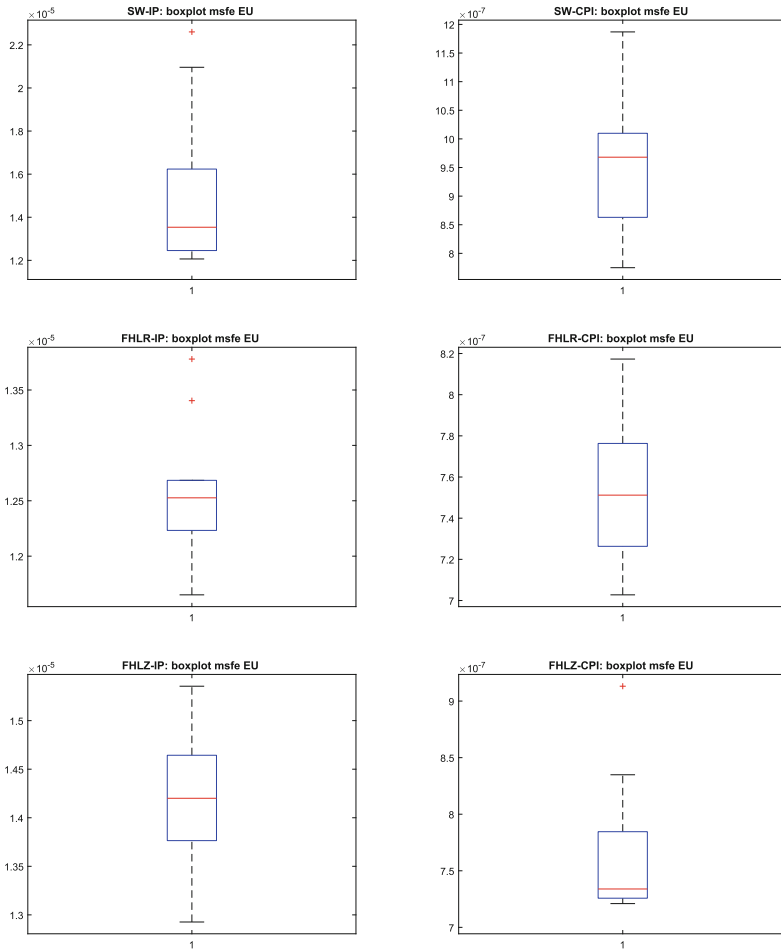


Fig. 1. Boxplot of the results on the EU dataset of the ten runs of the genetic algorithm for SW, FHLR and FHLZ over the IP (on the left) and over the CPI (on the right).

Since the results achieved for all dynamic factor models span a narrow region, we can conclude that the ten runs of the genetic algorithms for all methods have reached convergence. In addition, we can see that, over the IP, the dynamic methods show better results since the ten runs span a narrower region than

SW. Over the CPI, FHLZ shows better results since its ten runs of the genetic algorithm span a narrower region than the other methods. Moreover, the boxplot of FHLR covers a smaller region than SW.

3.6 Forecasting of the Industrial Production and the CPI

The forecasting performances of the three dynamic factor models over the IP and CPI are compared on the proper sample, which starts on January 2001 and ends on November 2015. The common benchmark for the factor models is the autoregressive process (AR) of order $p = 2$ for the IP and $p = 1$ for the CPI. However, as reported by CEPR, during the proper sample, the European economy faces two crisis periods: the first starts on May 2008 and ends on January 2009. The second starts on September 2011 and ends on March 2013. Hence, it is reasonable to assess whether the relative forecasting performances of the three dynamic factor models present a relevant change during the crisis periods. As in [2] and in [1], to assess the forecasting performance of each couple of methods locally, each time series of the dataset is smoothed by a centered moving average of length $m = 61$ (with coefficients equal to $1/m$) and then the Fluctuation test is run, at 5% significance level. Further details about this test can be found in [11]. The results for the IP at horizons $h \in \{6, 12, 24\}$ are reported in Fig. 2. All factor models outperform significantly the benchmark from the first crisis on at all horizons. SW tends to outperform the dynamic methods between the two crises. Instead, outside the period between the two crises, the dynamic methods show significantly better performances than SW. As to the performance of the dynamic methods, FHLR outperforms FHLZ between the two crises. To sum up, FHLR tends to outperform the other methods. However, this does not hold true in the period between the two crises, in which SW seems to be the most performing method. These results are similar to those obtained in [1], but in our exercise the relative performance of FHLR in comparison with SW are neater. The results for the CPI at horizons $h \in \{6, 12, 24\}$ are reported in Fig. 3. All methods perform better than AR significantly from the first crisis on. At horizon $h \in \{12, 24\}$, FHLR and FHLZ outperform SW on average on the whole sample, except between the two crises. As to the comparison of dynamic methods, at horizons $h \in \{6, 12\}$ FHLZ globally outperforms FHLR from the first crisis on. Instead, at horizon $h = 6$, FHLR globally outperforms FHLZ from the first crisis on. In comparison with [1], FHLR shows slightly better forecasting performance in comparison with other methods. In addition, SW seems to be the most performing method between the two crises.

FHLR and FHLZ tend to outperform SW at all horizons, except FHLR at horizon $h = 6$ during the first crisis. FHLR and FHLZ outperform AR at horizons $h \in \{6, 12\}$. At horizon $h = 24$, AR outperforms FHLR and FHLZ from the first crisis on. SW outperforms AR during the two crisis periods at horizons $h \in \{6, 12\}$. At horizons $h \in \{12, 24\}$, AR outperforms SW from the second crisis on. At all horizons, FHLZ outperforms FHLR during the first crisis. At horizons $h \in \{6, 12\}$, this behaviour seems to be persistent. Instead, at horizon $h = 24$, FHLR outperforms FHLZ from the second crisis on.

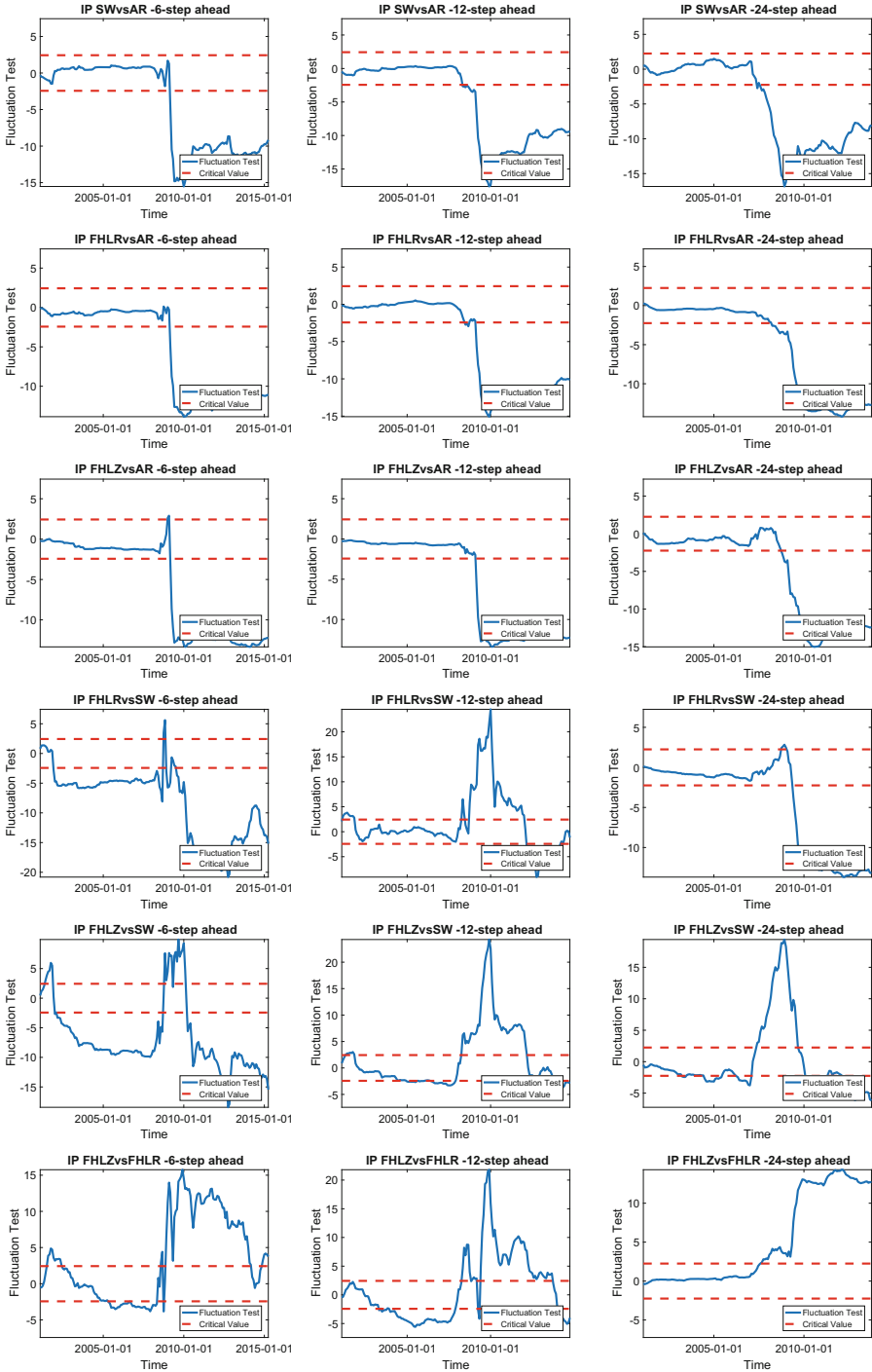


Fig. 2. Fluctuation test for the IP on the EU dataset.

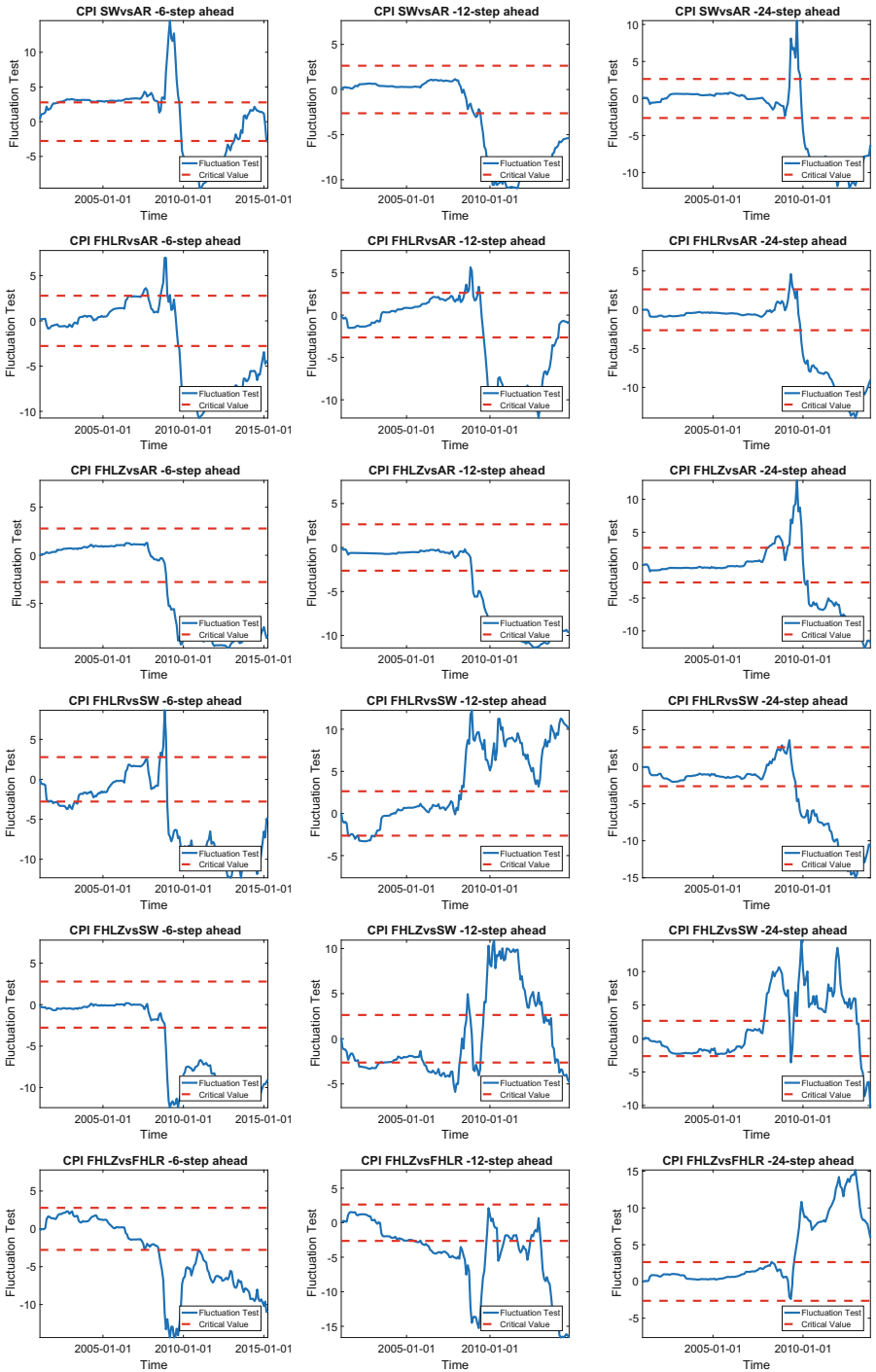


Fig. 3. Fluctuation test for the CPI on the EU dataset.

4 Results on the US Dataset

In this chapter, we will use the same notation as in [2].

4.1 Calibration of SW Model

As in Subsect. 3.1, by selecting the values of the parameters which guarantee the lowest mean rMSFE, the chosen configuration for the IP is the following:

$$(r, \alpha, \beta, l) = (BN, 0, 0, 12). \quad (4.1)$$

Instead, the chosen configuration for the CPI is the following:

$$(r, \alpha, \beta, l) = (3, 1, 10, 15). \quad (4.2)$$

4.2 Calibration of FHLR Model

As in Subsect. 3.2, by selecting the values of the parameters which guarantee the lowest mean rMSFE, the chosen configuration for the IP is the following:

$$(r, q, k, d, l) = (9, 2, Exponential, 40, 12). \quad (4.3)$$

Instead, the chosen configuration for the CPI is the following:

$$(r, q, k, d, l) = (6, 1, Hann, 25, 15). \quad (4.4)$$

4.3 Calibration of FHLZ Model

As in Subsect. 3.3, by selecting the values of the parameters which guarantee the lowest mean rMSFE, the chosen configuration for the IP is the following:

$$(q, k, d, ml, l) = (5, Triangular, 40, 2, 12). \quad (4.5)$$

Instead, the chosen configuration for the CPI is the following:

$$(q, k, d, ml, l) = (5, Triangular, 25, 5, 15). \quad (4.6)$$

4.4 Calibration of the Benchmark

As in Subsect. 4.4, by selecting the values of the parameter p which guarantee the lowest mean rMSFE, the chosen configuration for the IP is the following:

$$p = 2 \quad (4.7)$$

Instead, the chosen configuration for the CPI is the following:

$$p = 9 \quad (4.8)$$

4.5 Empirical Proof of the Convergence of the Runs of the Genetic Algorithm

To give an empirical proof of the convergence of the genetic algorithm, in Fig. 4 the boxplots of the results of the ten runs of each selected dynamic factor model for the IP (on the left) and for the CPI (on the right) are reported.

Since the results achieved for all dynamic factor models span a narrow region, we can conclude that the ten runs of the genetic algorithms for all methods over IP and over CPI have reached convergence. We can see that, over both the IP and the CPI, FHLZ shows better results since its ten runs of the genetic algorithm span a narrower region than the other methods. Moreover, the boxplot of FHLR covers a smaller region than SW.

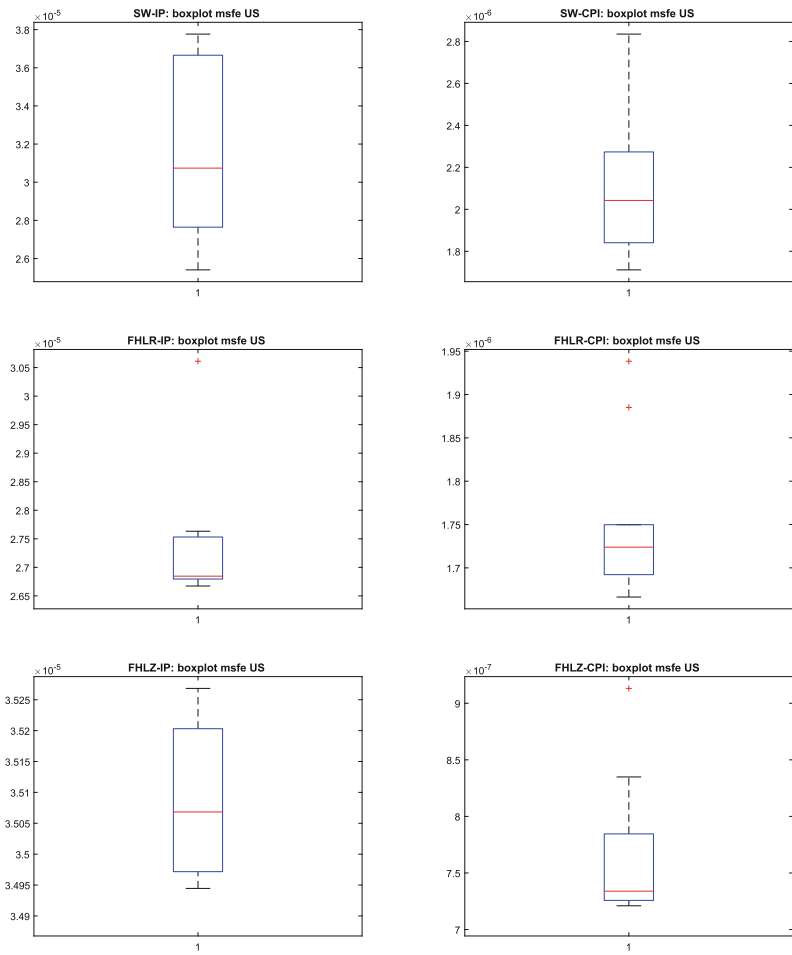


Fig. 4. Boxplot of the results on the US dataset of the ten runs of the genetic algorithm for SW, FHLR and FHLZ over the IP (on the left) and over the CPI (on the right).

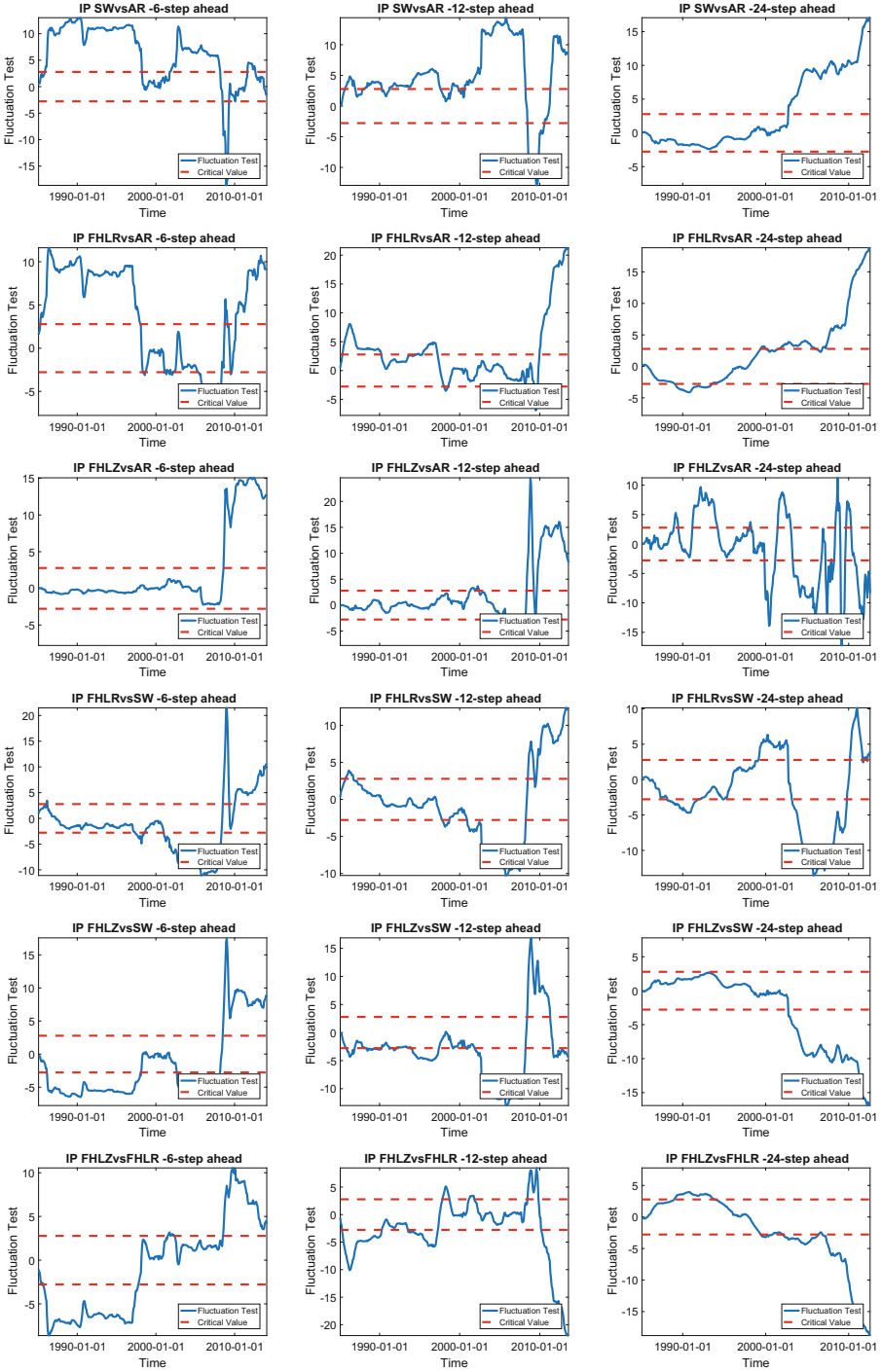


Fig. 5. Fluctuation test for the IP on the US dataset.

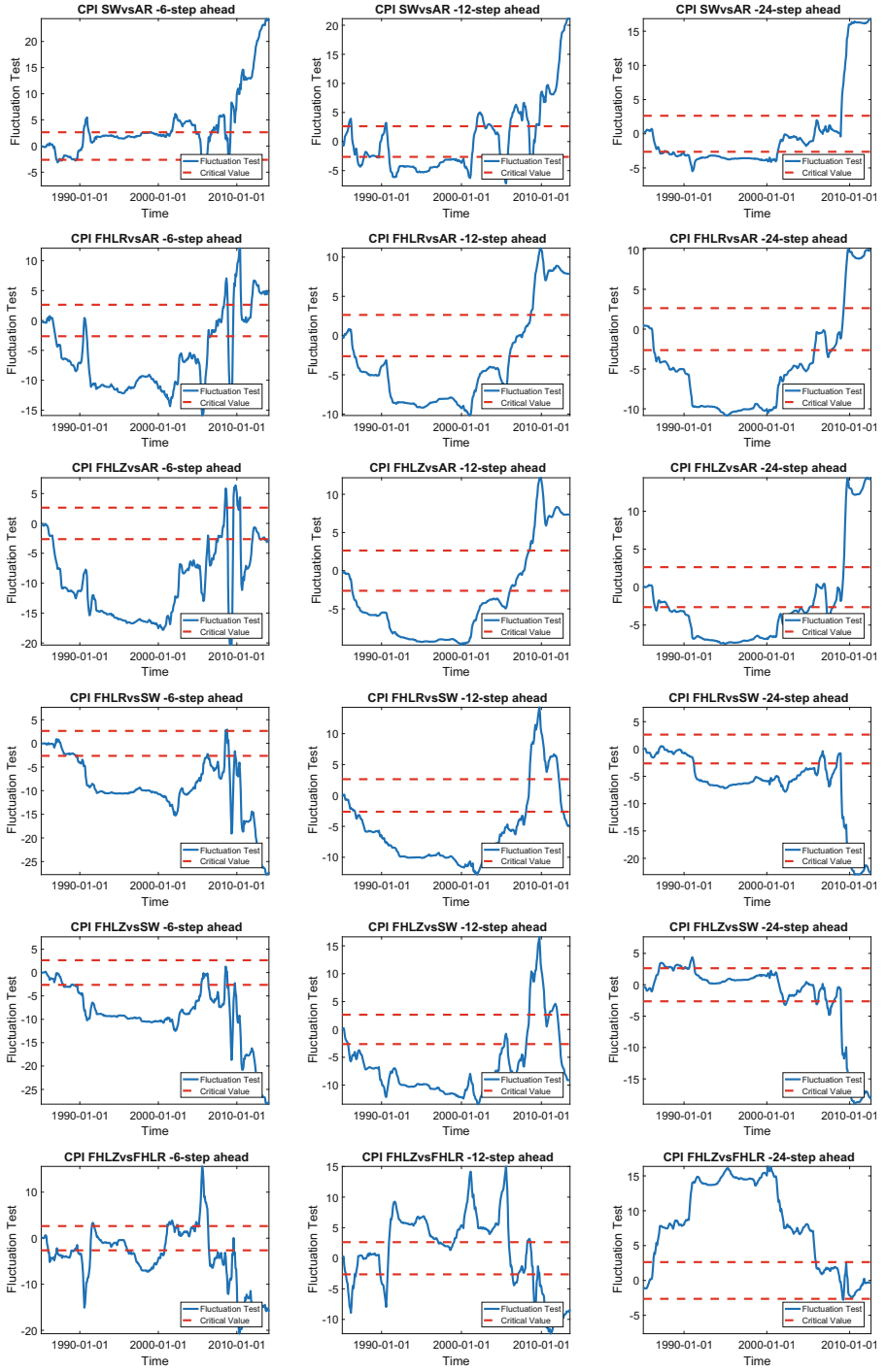


Fig. 6. Fluctuation test for the CPI on the US dataset.

4.6 Forecasting of the Industrial Production and the CPI

The forecasting performances of the three dynamic factor models over the IP and CPI are compared on the proper sample, which starts on March 1960 and ends on October 2014. The common benchmark for the factor models is the autoregressive process (AR) of order $p = 2$ for the IP and $p = 9$ for the CPI. However, as reported by NBER, during the proper sample, the American economy faces a crisis period which starts on December 2007 and ends on June 2009. Hence, it is reasonable to assess whether the relative forecasting performances of the three dynamic factor models present a relevant change during the crisis period. As in Subsect. 4.6, to assess the forecasting performance of each couple of methods locally, each time series of the dataset is smoothed by a centered moving average of length $m = 61$ (with coefficients equal to $1/m$) and then the Fluctuation test is run, at 5% significance level. The results for the IP at horizons $h \in \{6, 12, 24\}$ are reported in Fig. 5. The benchmark globally shows significantly better results than the factor models from the Great Recession on. However, this does not hold for SW at horizons $h \in 6, 12$ and for FHLZ at horizon $h = 24$ during the Great Recession. SW tends to outperform the dynamic methods from the Great Recession on, apart from FHLZ at horizon $h = 24$. As in [2], FHLR outperforms FHLZ from the Great Recession on. The results for the CPI at horizons $h \in \{6, 12, 24\}$ are reported in Fig. 6. No factor model seems to perform better than the benchmark from the Great Recession on. Instead, before the Great Recession, the contrary seems to hold. Dynamic methods show significant better performances than SW at all horizons, except for $h = 12$. FHLZ outperforms FHLR outside the Great Recession. Apart from this period, mixed evidences appear as far as the comparison between dynamic methods is concerned. Hence, as to the performances of dynamic methods, we can draw less clear conclusions than in [2].

5 Concluding Remarks

In this paper, we address the problem of calibrating dynamic factor models for macroeconomic forecasting. The novelty in this study consists in having designed and built a genetic algorithm for calibration. In this paper, we have empirically shown that the genetic algorithm in the calibration process plays a crucial role in this study, since a more efficient exploration of the parameter space allows us to empirically prove the superiority of frequency-domain dynamic factor models against time-domain factor model in a macroeconomic forecasting setting. We also notice that the time-domain factor model performs much better than the frequency-domain models considered in this paper. We eventually stress that our novel calibration approach has produced very good results in prediction.

Acknowledgements. We would like to thank Alessandro Giovannelli, Viviana Doldi, Marco Lippi, Valentina Mameli, Irene Poli, Simona Sanfelici, Debora Slanzi, Stefano Soccorsi and three anonymous referees for their support and comments on an earlier version of the manuscript.

References

1. Della Marra, F.: A forecasting performance comparison of dynamic factor models based on static and dynamic methods. *Commun. Appl. Ind. Math.* **8**(1), 44–66 (2017)
2. Forni, M., Giovannelli, A., Lippi, M., Soccorsi, S.: Dynamic factor model with infinite dimensional factor space: forecasting. CEPR discussion paper (2017)
3. Stock, J.H., Watson, M.W.: Forecasting using principal components from a large number of predictors. *J. Am. Stat. Assoc.* **97**(460), 1167–1179 (2002)
4. Stock, J.H., Watson, M.W.: Macroeconomic forecasting using diffusion indexes. *J. Bus. Econ. Stat.* **20**(2), 147–162 (2002)
5. Forni, M., Hallin, M., Lippi, M., Reichlin, L.: The generalized dynamic factor model: identification and estimation. *Rev. Econ. Stat.* **82**(4), 540–554 (2000)
6. Forni, M., Hallin, M., Lippi, M., Reichlin, L.: The generalized dynamic factor model: one-sided estimation and forecasting. *J. Am. Stat. Assoc.* **100**, 830–840 (2005)
7. Forni, M., Hallin, M., Lippi, M., Zaffaroni, P.: Dynamic factor model with infinite dimensional factor space: representation. *J. Econom.* **185**, 359–371 (2015)
8. Forni, M., Hallin, M., Lippi, M., Zaffaroni, P.: Dynamic factor model with infinite dimensional factor space: asymptotic analysis. In: EIEF Working Paper (2016)
9. Kapetanios, G.: Variable selection in regression models using nonstandard optimisation of information criteria. *Comput. Stat. Data Anal.* **52**(1), 4–15 (2007)
10. Kapetanios, G., Marcellino, G.M., Papailias, F.: Variable selection for large unbalanced datasets using non-standard optimisation of information criteria and variable reduction methods. In: Quantf Working Paper (2014)
11. Giacomini, R., Rossi, B.: Forecast comparisons in unstable environments. *J. Appl. Econom.* **25**(4), 595–620 (2010)



Functional Interactions in Complex Networks: A Three-Step Methodology for the Implementation of the Relevance Index (RI)

Riccardo Righi^(✉) and Sofia Samoili

European Commission, Joint Research Centre (JRC),
Unit B6 - Digital Economy, Seville, Spain
{riccardo.righi,sofia.samoili}@ec.europa.eu

Abstract. In order to enable the management of the large presence of similar groups of agents, namely masks, resulting from the implementation of the Relevance Index (RI) algorithm, the ‘PoSH-CADDY’ three-step methodology is here proposed. The developed procedure is based on (i) several rounds of analysis to be performed over reducing sets of agents (with a Progressive Skimming procedure), (ii) the consideration of the overlaps among masks emerging from the output of each round (by means of a Hierarchical Cluster Analysis), (iii) a final analysis of the masks remaining from the previous steps (by considering those with a minimum Degree of Dissimilarity). The methodology is implemented in a real socio-economic complex network. Insights from a first explorative analysis are provided.

Keywords: Functional interactions · Physical order
Relevance Index · Progressive skimming · Hierarchical clustering

1 Introduction

Since the widespread use of network analysis in mid 90’s [1], social sciences have mostly focused on the comprehension of the structure of durable relationships (e.g. friendship) and its evolution. However, in some contexts the concept of connections is required to represent something that is more similar to a series of flickering and dynamic interactions, than to stable relationships¹. When dynamic interactions are observed, the presence and the evolution of meso-structures²

The views expressed are purely those of the author and may not in any circumstances be regarded as stating an official position of the European Commission.

¹ There are cases in which, even if any new relationship is established, flickering interactions occur: people daily exchange messages with long-time friends, and enterprises repeatedly collaborate with partners they already know.

² In the present work, the concepts of (i) masks, (ii) groups, (iii) communities, or (iv) meso-structures, are all treated indistinctly since they all refer to subset of agents belonging to the same system.

can be scarcely investigated through the use of methods that are suitable for a process of stepwise creation/dissolution of connections [2]. The continuous activation/inactivation of links between agents demands the use of methodologies that, instead of considering the statistical significance of the formation/modification of the relational architectures, focus on the physical order contained in the occurred phenomena [3]. One example of a methodology that allows this is the Relevance Index algorithm, henceforth RI [4–6]. The RI, in order to investigate emergent temporal patterns in dynamic complex systems, uses a statistical approach to evaluate the significance of the integration in terms of entropy of agents' joint behaviors.

Although in complex networks analyses the detection of groups of agents is typically performed by focusing on agents' similarity or through the analysis of the network structure [7, 8], the creation and the implementation of the RI algorithm provides a new approach for community detection analysis. With the RI algorithm researchers can detect groups of agents characterized by high levels of behavioral integration. These behaviors, being significantly far from randomness, are expected to reveal the presence of a common function jointly pursued by all the involved members. Since low levels of entropy are determined by the repetition of specific combinations of joint individual statuses over time, the emergence of a non-random temporal pattern unveils the alignment of the actions of these individuals towards a common function. Nevertheless, to implement the RI algorithm in dynamic complex networks that have at least some thousands of agents, and that are observed in a number of instants that is sensibly lower than the number of agents involved, additional methodological steps have to be developed. In particular, in the present work the three-step 'PoSH-CADDy' methodology is developed so as to provide a possible solution to refine the presence of redundancy in the results provided by the RI algorithm when implemented on temporal networks having the aforementioned characteristics.

In Sect. 2 in which an overview of the proposed methodology is presented. In Sect. 3 the principles of the RI algorithm are introduced. Then, in Sect. 4, the first step of the methodology is described, regarding the run of the RI algorithm several times over sets of agents progressively reducing. Then, in Sect. 4 the second step of the methodology is described, regarding the implementation of a hierarchical agglomerative cluster analysis over masks, i.e. subsets of agents belonging to the analyzed system, detected at the previous step. Section 5 follows with the third and last step of the methodology, regarding a final treatment for redundancy of masks detected in all round. Because of this last step, a final set of masks, i.e. a partition of the system, is detected. Finally, in Sect. 6 the implementation of the methodology in a case study is presented. After selecting combinations of the introduced parameters, explorative considerations are made on partitions selected according to (i) a principle of maximization of the overall percentage of agents involved in the partition³ and (ii) a principle of minimization of the percentage of agents that belong to more than one mask.

³ Not necessarily all agents belong to at least one masks/subset.

2 Overview of the Methodology

Acknowledging other ongoing researches with similar objectives [9,10], the present work addresses the issue of redundancy that arises by implementing RI methodology in systems with a small ratio between the number of instants in time over which agents can be observed, and the number of agents involved. More specifically, the methodology aims to identify a limited number of masks of agents, i.e. subsets of agents detected by the RI algorithm, so as to allow a final simple representation of the functional meso-structures that are present in the considered complex network. The proposed methodology is based on the following three parameters:

- (1) R , i.e. the number of RI rounds of analysis that are performed,
- (2) v_{OV} , i.e. a threshold used as reference to limit the presence of overlapping agents among the subset of masks finally considered from each round of RI analysis,
- (3) v_{SM} , i.e. a second threshold used as reference to reduce redundancy among the masks remaining after all the previous steps.

Each parameter has a strong connection with one of the steps of the methodology: R parameter defines the length of a process of Progressive Skimming, based on the reiteration of the RI algorithm in rounds of analysis in which the best mask obtained in the previous round is dropped; v_{OV} parameter defines the development of a Hierarchical Cluster Analysis of the masks detected in each round; v_{SM} parameter defines the final process of refinements in which the remaining masks after all rounds are analyzed in terms of their Degree of Dissimilarity. The methodology is named ‘PoSH-CADDy’ and is summarized in Pseudo-Code 1. The refinement of the output of the RI analysis is performed attempting (i) a spread and wide exploration of the meso-structures of the system under analysis through progressive skimming, and moving towards the detection of masks that (ii) are the most significant (in terms of integration of the behaviors of the agents belonging to them), and that (iii) produce a limited degree of overlaps among them, so as to favor simplicity in the analysis of complex network’s dynamics. The ‘PoSH-CADDy’ procedure (independently from the RI algorithm) is implemented with the R language with a CPU Intel Core i5 2.6 GHz processor and 8 GB RAM. The computational time (with $R=24$ and where v_{OV}, v_{SM} is tested with 21 different values each) is approximatively of 5 h. This time period is essentially required for the computation of the distance matrices that are needed to implement the cluster analysis of each group of 15.000 masks that are detected by the RI algorithm in each round. The other steps require a computational time of some minutes. The work does not take into consideration the computational performance of the RI algorithm, as what developed applies to a procedure of refinements of its results.

Pseudo-code 1: ‘PoSH-CADDy’ methodology for implementation of the RI algorithm over a system $A = \{a_1, a_2, \dots, a_n, \dots, a_N\}$, where a_n is the n -th agent

```

function PoSH-CADDY ( $R \in \mathbb{N}^+$ ,  $v_{OV} \in \mathbb{R}_{\geq 0 \wedge \leq 1}$ ,  $v_{SM} \in \mathbb{R}_{\geq 0 \wedge \leq 1}$ )
  for each  $r$  round of analysis, where  $r \in \mathbb{N}^+$  and  $r \leq R$ , do
    Skimming of the best mask detected in the previous round
    Definition of  $A_r \subseteq A$  including the agents considered in the new round
    Detection of the set of masks  $\mathcal{O}(A_r)$  by means of RI analysis over  $A_r$ 
    for each possible number of clusters, i.e.  $\kappa \in \mathbb{N}^+$ , in which to split  $\mathcal{O}(A_r)$ , do
      Hierarchical agglomerative cluster analysis for binary data
      (with simple matching coefficient and complete linkage method)
      for each obtained  $k$ -th cluster, i.e.  $\mathbb{C}_{k,\kappa}(A_r)$ , do
        Selection of the mask with the highest  $t_{CI}$ 
      end for
      Measurement of the resulting overlaps by means of  $s_{OV}(r, \kappa)$ , i.e. the ratio between
      the number of agents included in at least two of the remaining masks, and the number
      of agents included in at least one of the remaining masks
    end for
    Definition of the set  $\mathcal{K}_{r,v_{OV}}$ , including those  $\kappa$  such that  $s_{OV}(r, \kappa) \leq v_{OV}$ 
    Selection of  $\bar{\kappa}_{r,v_{OV}}$ , i.e. the highest value of  $\kappa \in \mathcal{K}_{r,v_{OV}}$ 
    Definition of set  $\mathbb{P}_{r,v_{OV}}$ , by finally considering the results of the introduced cluster analysis
    with a number of clusters equal to  $\bar{\kappa}_{r,v_{OV}}$  and considering for each cluster only the mask
    with the highest  $t_{CI}$ 
    end for
    Definition of the unique set of masks  $\mathcal{P}_{R,v_{OV}}$ , including all the  $\mathbb{P}_{r,v_{OV}}$ 
    Definition of  $\mathbb{P}_{R,v_{OV}}^+$ , by sorting the masks in  $\mathcal{P}_{R,v_{OV}}$  in decreasing order of  $t_{CI}$ 
    Computation of dissimilarity between all masks in  $\mathbb{P}_{R,v_{OV}}^+$ 
    for each couple of masks having a Jaccard index  $> v_{SM}$  do
      Drop of the mask with the lower  $t_{CI}$ 
    end for
    Definition of the final set  $\mathcal{F}_{R,v_{OV},v_{SM}}$ , including the remaining masks
  end function

```

3 Principles of the RI Algorithm

The Relevance Index algorithm takes its origin from the neurological studies of Giulio Tononi in the 90’s. Tononi introduced the notion of functional cluster, defining it as *a set of elements that are much more strongly interactive among themselves than with the rest of the system, whether or not the underlying anatomical connectivity is continuous* [11]. The hypothesis was confirmed as neurons with similar functions are found to demonstrate high level of coordination in their behaviors over time, independently from being (or not) situated in the same brain region [12,13]. The Cluster Index (henceforth, CI), i.e. the statistics developed and tested by Tononi in his work [12], is based on two information theory concepts derived from the Shannon entropy: Integration (I) and Mutual Information (MI). Formally, given the set $A = \{a_1, a_2, \dots, a_n, \dots, a_N\}$ made of N agent and a mask of agents B^m such that $B^m \subset A$, the CI of B^m is written as follows

$$CI(B^m) = I(B^m)/MI(B^m, A \setminus B^m) \quad (1)$$

where $2 \leq |B^m| < |A|$ and $0 < m \leq \xi$, with $m \in \mathbb{N}^+$ and $\xi \approx 2^{|A|}$.

Since integration and mutual information values depend on the size of the subsystem that is under analysis, a homogeneous system made of variables having the same probabilities of the variables of the original system, but that do

not have correlation⁴ is used [4,5,12]. Finally, the level of significance of the normalized CI, namely t_{CI} , is the value according to which the final ranking of the subsets is produced:

$$CI'(B^m) = \frac{I(B^m)}{\langle I_h \rangle} / \frac{MI(B^m, A \setminus B^m)}{\langle MI_h \rangle} \quad (2)$$

$$t_{CI} = \frac{CI'(B^m) - \langle CI'_h \rangle}{\sigma(CI'_h)} \quad (3)$$

where $\langle I_h \rangle$ and $\langle MI_h \rangle$ indicate respectively the average integration of subsets of dimension $|B^m|$ belonging to the homogeneous system, and the average mutual information between these subsets and the remaining part of the homogeneous system. $\langle CI'_h \rangle$ and $\sigma(CI'_h)$, respectively the mean and the standard deviation of normalized cluster indices of subsets that have the same size of B^m and that belong to the homogeneous system, are used to compute the statistical index t_{CI} .

The concept of CI and t_{CI} was introduced in the research areas of artificial network models, of catalytic reaction networks and of biological gene regulatory systems, contributing to the identification of emergent meso-level structures [4]. Since an exhaustive computation of the t_{CI} statistic is possible only in small artificially designed networks, as those that were initially used to test the efficacy of the method [4–6], a genetic algorithm aimed to investigate the relevant subsets was implemented [6] in the RI algorithm. When implemented in large systems that can be observed in a relatively small number of instants in time, the RI algorithm produces a large number of possible B^m , which may differ among them just for the presence/absence of a single agent. As many similar masks are detected, redundancy emerges.

4 Step 1: Progressive Skimming of the Best Mask Detected

In order to address the large presence of similar masks detected in the considered system $A = \{a_1, a_2, \dots, a_n, \dots, a_N\}$ made of N agent, the first step that is proposed is the run of several rounds of the RI algorithm. Each round $r \in \mathbb{N}^+$, with $r \leq R$, and where $R \in \mathbb{N}^+$ indicates the number of rounds finally performed, is set to produce the detection of a same number of masks⁵. At the

⁴ A homogeneous system is a system having the same number of agents of the system to which it is referred; each agent has a random generated behavior in accordance with the probability of the states it assumes in the reference system.

⁵ The fact that the number of masks detected does not change, is just a choice of the researcher. This parameter could change but, since this work is not aimed at considering the increasing of the value of M , which has been fixed equal to 15.000 in each round of analysis, M is taken for given. Because of that, M will not be indexed with the number of the round r .

same time, in each r round a different set of agents is considered, namely $A_r = \{a_1, a_2, \dots, a_{n_r}, \dots, a_{N_r}\}$ where $A_r \subset A$ and with $|A_r| = N_r$. In order to formally describe the output of any round r of the analysis, the sets of masks detected by the RI algorithm, namely $\mathbb{O}(A_r)$, is defined according to the corresponding round of analysis. Formally,

$$\mathbb{O}(A_r) = \{B_r^1, B_r^2, \dots, B_r^m, \dots, B_r^M\} \quad (4)$$

where

- i. $B_r^m = \{a_{n_r} \in A_r : b_{m,n_r} = 1\}$ is the m -th mask detected
- ii. $b_{m,n_r} = \begin{cases} 1, & \text{if the agent } a_{n_r} \text{ is detected in the } m\text{-th mask} \\ 0, & \text{otherwise.} \end{cases}$
- iii. $t_{CI}(B_r^m) \geq t_{CI}(B_r^{m+1})$

For the definition of each different set of agent A_r , a cascade process is used. Before each round, the agents belonging to best mask detected in the previous round, i.e. B_{r-1}^1 , are dropped from the analysis, such that the cardinality of the set of agents considered, i.e. A_r , decreases after each round. Formally, each A_r can be described as

$$A_r = A \setminus \bigcup_{q=0}^{r-1} B_q^1 \quad (5)$$

where $q \in \mathbb{N}$ indicates one of the rounds preceding the r -th round⁶, and where $0 \leq q \leq (R - 1)$. Therefore, $A_r \subset A_{r-1} \forall r$. As in each round, the set that is analyzed with the RI algorithm does not include any of the best masks detected in the previous rounds, this procedure is called ‘progressive skimming’.

The RI algorithm produces a list of ordered binary masks that may differ among them just for the presence/absence of one single agent. Therefore, for the best detected mask of agents, i.e. B_r^1 , also many similar masks are detected (as they are likely to perform well also from a point of view of entropy) in $\mathbb{O}(A_r)$. Because of this redundancy⁷ the progressive skimming of masks is implemented, so as to perform an extended exploration of the system. This procedure, even if it deals with a loss of information and a reduction (and so also change) of the considered system when continuing the analysis round after round, allows the researcher to analyze how the rest of the system works independently from what in the previous rounds has been detected the group of agents with the most integrated behaviors. Interactions between the best mask detected in round r and masks detected in following rounds are limited, since the agents belonging to B_r^1

⁶ Since the initial round that is performed is $r = 1$, if $r = 1 \rightarrow q = 0$. As there is no round 0, if $q = 0 \rightarrow B_0^1 = \emptyset$. Therefore, from Eq. 5, when $r = 1$, we have that $A_1 = A \setminus B_0^1 = A \setminus \emptyset = A$.

⁷ Furthermore, the problem of redundancy in $\mathbb{O}(A_r)$ does not affect only the best mask B_r^1 . It is important to remark that it is also present for masks different from the best one. Therefore, it can be said that when the system is large, in each $\mathbb{O}(A_r)$ a lack of variety comes up.

are removed from the sets of agents that is going to be analyzed in the rounds following the r -th. However, because of the implementation of a hierarchical agglomerative cluster analysis (Sect. 5), in each round r also all the other masks different from B_r^1 are taken into account. Therefore, the progressive skimming does not imply that the best mask B_r^1 stands in a condition of isolation. If mask B_r^1 has significant intersections/interactions with other masks B_r^m detected in the same round, evidences should appear in the cluster analysis of the whole $\mathcal{O}(A_r)$. In contrary, if masks substantially different from B_r^1 do not emerge from the cluster analysis, some clues of a functional detachment between the agents in B_r^1 and the agents that belong to the rest of the system are detected.

5 Step 2: Clusters of Masks Within Each r -th Round

5.1 The Cluster Analysis of Masks in $\mathcal{O}(A_r)$

With the Simple Matching Coefficient (SMC) distance is measured between couples of masks, and with the Complete Linkage (CL) criterion for the progressive merging of clusters, a hierarchical agglomerative cluster analysis is then implemented. This analysis is here represented by the function θ_κ assigning each mask B_r^m to one (and only one) cluster. Formally,

$$\theta_\kappa^{SMC,CL}(B_r^m) = k \tag{6}$$

where $k \leq \kappa$, with $k \in \mathbb{N}^+$ indicating the specific cluster to which each mask $B_r^m \in \mathcal{O}(A_r)$ is assigned through the hierarchical cluster analysis (with SMC and CL) in which the masks of $\mathcal{O}(A_r)$ are allocated in a number of clusters equal to $\kappa \in \mathbb{N}^+$. Since the number of clusters is not established a-priori, at this stage the definition of each cluster, namely $\mathbb{C}_{k,\kappa}(A_r)$, has to take into account the fact that κ can vary. Therefore, each cluster $\mathbb{C}_{k,\kappa}(A_r)$ is formally defined as

$$\mathbb{C}_{k,\kappa}(A_r) = \{B_r^m \in \mathcal{O}(A_r) : \theta_\kappa(B_r^m) = k\} \tag{7}$$

where $\mathbb{C}_{k,\kappa}$ is the k -th cluster, obtained by dividing in κ clusters the masks contained in $\mathcal{O}(A_r)$.

5.2 The Selection of a Representative Masks for Each Cluster

For any cluster obtained, only the mask with the highest t_{CI} is considered, as representative of the cluster itself. Formally, this mask, henceforth indicated as $\tilde{B}_{r,k,\kappa}$, has the following properties:

$$\tilde{B}_{r,k,\kappa} \in \mathbb{C}_{k,\kappa}(A_r) \quad \text{and} \quad t_{CI}(\tilde{B}_{r,k,\kappa}) = \max t_{CI}(\mathbb{C}_{k,\kappa}(A_r)). \tag{8}$$

Therefore, each cluster is represented by the mask that, belonging to it, is also the one whose agents present a joint behavior that is the significantly farthest from randomness. By adopting this criterion, the principles underpinning the RI algorithm are respected. Even if several different combination may be present, the analysis of the similarity reveals groups of masks that have to be intended just as possible modification of the one of reference, i.e. the most relevant one.

5.3 Overlaps and the s_{OV} Statistic

The cluster analysis of $\mathcal{O}_r(A_r)$ and the selection of the mask with the highest t_{CI} for each cluster, can produce the affiliation of agents to more than one masks⁸. In order to set the value of κ , i.e. to determine the number of clusters, a criterion concerning the limitation of the progressive emergence of overlaps in the observed structure of masks is adopted. In order to understand which degree of overlap is associated with the values of κ , starting from 1 and continuing in increasing order, the statistic $s_{OV}(r, \kappa)$, where the subscript ‘OV’ stands for OVerlaps, is computed as

$$s_{OV}(r, \kappa) = \frac{|\bigcup_{k_\alpha, k_\beta=1}^{\kappa} (\tilde{B}_{r, k_\alpha, \kappa} \cap \tilde{B}_{r, k_\beta, \kappa})|}{|\bigcup_{k=1}^{\kappa} \tilde{B}_{r, k, \kappa}|} \quad \forall k_\alpha \neq k_\beta \quad (9)$$

where $k_\alpha, k_\beta \in \{1, \dots, k, \dots, \kappa\}$ are the indices of two distinct clusters $\mathcal{C}_{k, \kappa}(A_r)$, obtained by implementing the function θ_κ over the set of masks $\mathcal{O}(A_r)$. The statistic $s_{OV}(r, \kappa)$ calculates, for each possible value of r and of κ , the ratio between the number of agents that belong to at least two masks (numerator) and the number of agents that belong to at least one mask (denominator). The introduced statistic aims to evaluate the degree of simplicity associated to each possible number value of κ , i.e. the number of clusters in which to group the masks included in $\mathcal{O}(A_r)$. The simplicity lies on the fact that masks have to be recognizable and distinct from each other. If the structure of the detected masks is characterized by a high degree of overlap, the masks are so intertwined that they cannot be assumed as unitarity entities and the representation of the whole system, that they are suppose to provide, is finally unreadable.

5.4 Selection of the Number of Clusters by Means of v_{OV} Parameter

In order to define the value of κ , i.e. the number of clusters in which to split each set of masks $\mathcal{O}_r(A_r)$, the criterion adopted lies in the comparison between the statistic $s_{OV}(r, \kappa)$, defined by Eq. 9, and a percentage threshold used as reference, namely $v_{OV} \in \mathbb{R}_{\geq 0}$, with $0 \leq v_{OV} \leq 1$. Given a specific value of v_{OV} , the value κ is chosen in order to have the highest number of clusters among those to which corresponds a $s_{OV}(r, \kappa)$ lower than, or equal to, the percentage threshold v_{OV} . For each r -th round, a set of possible value of κ is so selected. These sets, namely $\mathcal{K}_{r, v_{OV}}$, are formally described as follows.

$$\mathcal{K}_{r, v_{OV}} = \{\kappa \in \mathbb{N}^+ : s_{OV}(r, \kappa) \leq v_{OV}\} \quad (10)$$

For each round r , depending on the threshold v_{OV} , all the values of κ that produce a partition for which the percentage of agents that belong to more

⁸ The allocation in one exclusive cluster does not concern agents. The same agent can be detected in two masks that are not included in the same cluster.

than one group (up to the number of agents overall included) is less or equal to the considered threshold v_{OV} , are considered admissible. Then, among all the elements contained in $\mathcal{K}_{r,v_{OV}}$, the value $\tilde{\kappa}_{r,v_{OV}}$, i.e. the final value in which finally to split the resulting masks contained in $\mathcal{O}_r(A_r)$ given the specific threshold v_{OV} , is defined as

$$\tilde{\kappa}_{r,v_{OV}} = \max \mathcal{K}_{r,v_{OV}} \quad (11)$$

By identifying $\tilde{\kappa}_{r,v_{OV}}$, the highest number of cluster, given the threshold v_{OV} , is selected. Therefore, the soft partition⁹ obtained in any of the r rounds, namely $\mathbb{P}_{r,v_{OV}}$, and can be formally defined as

$$\mathbb{P}_{r,v_{OV}} = \{\tilde{B}_{r,k,\kappa} \in \mathcal{O}(A_r) : \kappa = \tilde{\kappa}_{r,v_{OV}}\} \quad (12)$$

6 Step 3: Final Treatment of Redundancies

6.1 The Set of Masks Resulting from All the Rounds: $\mathcal{P}_{R,v_{OV}}$

At the end of an entire process of analysis always¹⁰ with the same value of the parameter v_{OV} , R sets of masks are obtained, and each of them is identified by the corresponding $\mathbb{P}_{r,v_{OV}}$. Therefore, since the analysis is developed with a specific value of R and a specific value of v_{OV} , it is possible to assemble all the masks in a unique set, namely $\mathcal{P}_{R,v_{OV}}$, that can be formally defined as

$$\mathcal{P}_{R,v_{OV}} = \{\tilde{B}_{r,k,\kappa} \in \bigcup_{r=1}^R \mathcal{O}(A_r) : \kappa = \tilde{\kappa}_{r,v_{OV}}\} \quad (13)$$

where $\tilde{\kappa}_{r,v_{OV}}$ is the number of clusters in which the specific $\mathcal{O}(A_r)$ is divided, as a result of the process described in Eqs. (8–11), and where, as explained in Eq. (8), the tilde ($\tilde{\cdot}$) over the mask $B_{r,k,\kappa}$ indicates that in the cluster of masks to which it belongs, i.e. $\mathcal{C}_{k,\kappa}(A_r)$, the mask $\tilde{B}_{r,k,\kappa}$ presents the highest t_{CI} . Once the set $\mathcal{P}_{R,v_{OV}}$ is defined, the last issue addresses the consequence of having implemented a reiterated procedure of analysis, i.e. multiple rounds of the RI algorithm. As at the beginning of each round r exclusively the agents belonging to B_{r-1}^1 are dropped, the presence of similar masks (among all those detected in an entire process of analysis) is not prevented¹¹. The following, and last, steps aim to manage this redundancy.

⁹ A soft partition is intended to be a set of masks of agents that do not necessarily belong to exclusively one masks. Therefore, as explained above, an agent can belong to more than one mask.

¹⁰ From the first round $r = 1$, to the last round $r = R$.

¹¹ The set $\mathcal{P}_{R,v_{OV}}$ can present redundancies since, even if the rest of the system that at each new round r is analyzed does not include the best masks detected in round $(r-1)$, it can include the agents that belong to the second/third/etc. masks detected in the round $(r-1)$. Therefore, it could happen that those masks that were detected as second/third/etc. masks in $(r-1)$, are detected also in the round r .

6.2 Sorting the Masks of $\mathcal{P}_{R,v_{OV}}$ in Decreasing Order of t_{CI}

All the masks belonging to $\mathcal{P}_{R,v_{OV}}$ are sorted in decreasing order, according to the value of their t_{CI} . In this way, from the set of masks $\mathcal{P}_{R,v_{OV}}$, the sorted set of masks $\mathcal{P}_{R,v_{OV}}^+$ is generated. Formally,

$$\mathcal{P}_{R,v_{OV}}^+ = \{\tilde{B}_{R,v_{OV}}^{(1)}, \tilde{B}_{R,v_{OV}}^{(2)}, \dots, \tilde{B}_{R,v_{OV}}^{(j)}, \dots, \tilde{B}_{R,v_{OV}}^{(J)}\} \quad (14)$$

where $|\mathcal{P}_{R,v_{OV}}| = J$, and $\tilde{B}_{R,v_{OV}}^{(j)}$ is one of the masks belonging to $\mathcal{P}_{R,v_{OV}}$ and that were previously indicated as $\tilde{B}_{r,k,\tilde{k}_{r,v_{OV}}}$. Moreover, the index in the superscript, i.e. $j \in \mathbb{N}^+$ where $j \leq J$, refers to the ordinality of the masks of $\mathcal{P}_{R,v_{OV}}^+$, so that is true the condition $t_{CI}(\tilde{B}_{R,v_{OV}}^{(j)}) > t_{CI}(\tilde{B}_{R,v_{OV}}^{(j+1)})$.

6.3 Final Drop of Similar Masks According to the Paramater v_{SM}

Once the masks are ordered according to their t_{CI} , a final analysis of their similarity is performed. Starting from the best mask $\tilde{B}_{R,v_{OV}}^{(1)}$, all the masks that are too similar to it are dropped. Then, the same procedure is repeated in cascade process. The second best mask of those remaining is then compared with those having a lower t_{CI} , and so forth with the third best mask remaining, the fourth, etc. This procedure continues until there are no more masks that can be used as a reference. In this way, only masks that have a minimum degree of dissimilarity are kept.

The similarities between masks are calculated in terms of JaCcard Index¹² (henceforth, JC), i.e. the percentage of the number of agents in the intersection of the two considered masks (up to the number of agents in the union set of the same two masks). Then, the set of masks $\mathcal{P}_{R,v_{OV}}^+$ is filtered using a threshold regarding SiMilarity, namely $v_{SM} \in \mathbb{R}_{\geq 0}$, with $0 \leq v_{SM} \leq 1$. The resulting (and final) set of masks, namely $\mathcal{F}_{R,v_{OV},v_{SM}}$, can be formally defined as

$$\mathcal{F}_{R,v_{OV},v_{SM}} = \{\tilde{B}_{R,v_{OV}}^{(j)} \in \mathcal{P}_{R,v_{OV}}^+ : JC(\tilde{B}_{R,v_{OV}}^{(i)}, \tilde{B}_{R,v_{OV}}^{(j)}) < v_{SM}\} \quad (15)$$

where

- i. $\tilde{B}_{R,v_{OV}}^{(i)} \in \mathcal{P}_{R,v_{OV}}^+$,
- ii. i and j , where $i \in \mathbb{N}$ and $j \in \mathbb{N}^+$ and $0 \leq i < j$, are used to indicate the mask of $\mathcal{P}_{R,v_{OV}}^+$ by making reference to their ordinality, as described in Eq. 14,
- iii. $\tilde{B}_{R,v_{OV}}^{(0)} = \emptyset$, so that $JC(\tilde{B}_{R,v_{OV}}^{(0)}, \tilde{B}_{R,v_{OV}}^{(1)}) = 0$.

¹² While in Step 2 of the proposed methodology the SMC is used to evaluate similarity (see Sect. 5), in this Step the JC is considered as more appropriate. JC focuses its attention on the intersection of two masks (with regard of the union set), while SMC considers as a condition of similarity also the simultaneous absence of a same element. While in Step 2 was important to consider also the co-absence of agents as an element of similarity, so as to evaluate where the algorithm had moved (in terms of agents considered and not considered), here only the presence of overlapping agents, i.e. the intersection, is relevant.

7 Case Study - Region Tuscany Innovation Policies

In this Section, an example of an implementation of the RI+PoSH-CADdy methodology in an empirical analysis, is presented. The considered case study addresses a regional programme implemented by Tuscany Region (Italy) in the period 2000–2006, aiming to support innovation projects. The considered network policy programme sustained the development of innovation processes by fostering interactions between local agents (enterprises, universities, public research centers, local government institutions, service centers, etc.) [14–16]. Starting in 2002 (and ending in 2008), the programme of public policies was consisted of nine waves not uniformly distributed over time: they had different durations and they overlapped, producing periods in which no wave was active, and periods in which three waves were simultaneously active. The degree of formation and of dissolution of connections was so high that resulted in a situation of intense discontinuity over time. Therefore, a new appropriate tool that does not investigate the flourishing of communities looking at the stepwise creation of network frameworks, was deemed necessary [2]. Moreover, by using the RI algorithm the analysis could take into account the presence of functional meso-structures. Finally, because of the objective of policies taken into consideration, i.e. fostering of innovative processes, the focus on interactive dynamics, more than on network’s relational architectures, is even more meaningful¹³ [17].

7.1 Available Data and Pre-processing

The most important aspect regarding the implementation of RI analysis in the present case study regards the definition of the informational basis describing agents’ statuses of activity. Since the available data contains information on the starting and the ending dates of agents’ participations in the projects, it is possible to define a set of 59 instants in time¹⁴ to observe the system. With these dates, a complete behavioral profile for each of the agents involved in the policy programme is structured. In each instant, the number of projects in which each agent was active is considered. A series of 58 variables is generated taking into account how the levels of activity vary from one instant to the following

¹³ In this case study, the agents’ activities coincide with interactions. Agents are considered to be active when they are participating in a project. And since in each project partnerships have to be established (no single-participant projects are allowed), it follows that to be active implies to be interacting.

¹⁴ Considering all the dates of starting and the ending of the projects, 59 different dates were identified.

one¹⁵. Regarding the size of the system, the agents participating in the described policies of Region Tuscany are 1121, and the majority of them participated just in one project. The scarcely active agents are removed from the analysis. The focus is set on those with a minimum degree of activity. Therefore, only agents that at least participated in 2 projects are considered. Finally, 352 agents remain. These agents constitute the initial set of the analysis, namely A .

7.2 Setting the Parameters

The RI analysis with the PoSH-CADDy methodology is implemented over the set A , consisted of 352 agents, observed in 58 instants over time. The number of rounds to be performed, i.e. the parameter R , is set equal to 24. With the progressive skimming, as described in Sect. 4, A_{24} is consisted of 204 agents. Therefore, A is extensively explored, as the procedure is stopped after having removed the 45.17% of the agents initially involved. Regarding the threshold v_{OV} , since no specific theoretical reasons suggest the a-priori identification of a specific value, a discrete set V_{OV} of percentage thresholds is used and each $v_{OV} \in V_{OV}$ is considered to implement a process of analysis. The set V_{OV} is defined as follows:

$$V_{OV} = \{v_{OV} \in \mathbb{R}_{>0} : v_{OV} = \frac{1}{40} x\} \quad \forall 0 \leq x \leq 20, x \in \mathbb{N} \quad (16)$$

Regarding the setting of the threshold v_{SM} , the same set of conditions are applied also for v_{OV} . A discrete set V_{SM} is created in the same way of V_{OV} and each $v_{SM} \in V_{SM}$ is considered to implement the analysis. For both, values larger than 0.5 are not taken into consideration as, in principle, they go in the opposite direction of the general objective of the present work, that is to reduce redundancy¹⁶.

As one value for R , and 21 values for v_{OV} , and 21 values for v_{SM} are considered, 441 different $\mathcal{F}_{R,v_{OV},v_{SM}}$ are finally computed. Each of these final sets of RI masks constitute a soft partition¹⁷ of the system A . In Fig. 1a, the 441

¹⁵ These variables assume four different values that correspond to one of the following four situations: inactivity, decreasing activity, stable activity or increasing activity. The ‘activity’ status is defined by considering the number of projects in which the agent is participating in the corresponding instant, with regard to the number of projects in which it was participating in the previous instant. With these series of variables, a second order Markov condition is taken into account, since agents’ activity is not described just for what is in each instant, but for what it is in the present conditioned to what it was in its nearest past. As a variation in time is considered, the number of variables finally computed equals the number of variables initially present minus 1.

¹⁶ To have more than the 50% of agents producing an overlaps among the masks of a generic $\mathbb{P}_{r,v_{OV}}$, or to allow in $\mathcal{F}_{R,v_{OV},v_{SM}}$ couples of masks generating an intersection that is the 50% or more of the corresponding union set, has been considered as not pertinent for the objective of this work.

¹⁷ Overlaps among groups (determined by the fact that each agent can belong in more than one group) are allowed and are present.

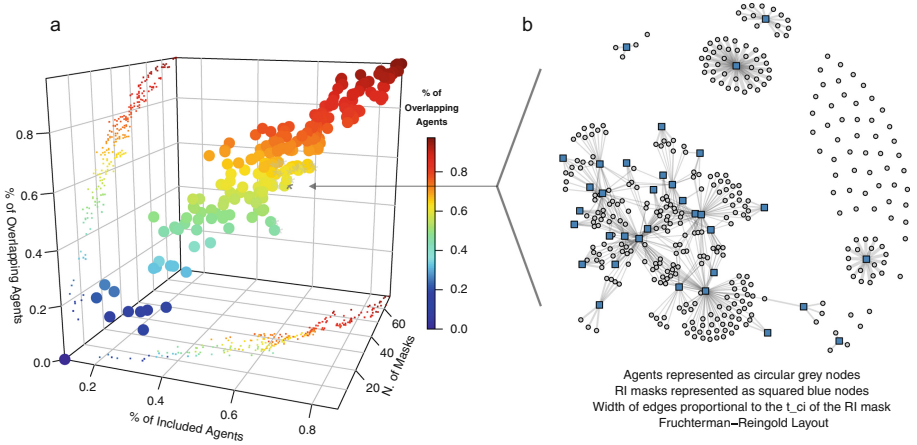


Fig. 1: (a): Colored dots represent the final partitions obtained, i.e. all the $\mathcal{F}_{R,v_{OV},v_{SM}}$ resulting from the possible combinations of the three parameters $R = 24, v_{OV} \in V_{OV}$ and $v_{SM} \in V_{SM}$, as described in Eq. (16). The y -axis describes the percentage of agents that, in the corresponding $\mathcal{F}_{R,v_{OV},v_{SM}}$, belong to more than one mask (up to the number of agents that belong to at least one mask). The x -axis describes the percentage of agents that belong at least to one mask (up to the total number of agents included in the initial considered set A). The z -axis describes the number of masks that are present in the corresponding partition. The color of the 441 dots is in accordance with the % of overlapping agents. Small grey asterisks indicate the 47 partitions that include at least the 60% of agents of A , and that have less than the 70% of overlapping agents. Big colored dots are projected on the lateral and on the bottom faces of the cube delimiting the three-dimensional space. (b): Bipartite graph representing affiliations of agents (of set A) in the specific final set of RI masks $\mathcal{F}_{24,0.325,0.225}$ (indicated in the 3D representation on the left, with a darker grey asterisk). Grey circular nodes represent agents, and blue squared nodes represent RI mask. The width of edges is proportional to the t_{CI} of the mask. (Color figure online)

obtained $\mathcal{F}_{R,v_{OV},v_{SM}}$ are illustrated in a three-dimensional space describing the number of agents included (up to the total number of agents included in A), the percentage of agents belonging to more than one mask (up to the number of agents overall included in the partition), and the number of masks included in the corresponding partition.

7.3 Exploration of the Results

As represented in Fig. 1a, the considered combinations of the three parameters lead to different $\mathcal{F}_{R,v_{OV},v_{SM}}$. Even though currently evaluation on the parameters' space is not effectuated, the choice of the value to be considered is attributed with an a-posteriori unbiased procedure. In the present work, only those parti-

tions including (i) at least 60% of the agents of the initial set A , and (ii) having less than 70% of agents belonging to more than one community, are considered. The parameters' space is narrowed in order to address two objectives, which both concern the readability of the final representations of the system. These objectives are: (i) to consider partitions in which a large part of the initial system is analyzed, and (ii) to avoid the selection of partitions in which extreme overlapping of the detected subsets prevents a simple interpretation of the system. Statistics regarding the feature of the single masks are not taken into account, and a-priori biased considerations on the values of v_{OV} and v_{SM} are not made. Currently, the parameters' space is not explored with a standardized method. However, the parameters are not selected based on the properties of the single masks, so as to avoid bias.

Based on the aforementioned conditions, 47 partitions (up to 441) are identified. These partitions are indicated with grey asterisks in Fig. 1a. In order to proceed with the exploration of the first results provided by the methodology, the presence of similar features within all the groups of 47 partitions is suggested. Currently, only one is heuristically selected, namely the partition with $v_{OV} = 0.325$ and $v_{SM} = 0.225$, which is indicated with a black asterisk in Fig. 1a. The corresponding set of masks, i.e. the masks included in $\mathcal{F}_{24,0.3,0.075}$, is intended as a weighted bipartite graph, as represented in Fig. 1b. The agents involved, represented by grey circular nodes, are connected to the RI masks, represented by blue squared nodes, in which they are included, and the weight of their connection is based on the value of the t_{CI} of the masks¹⁸. This partition is composed by 34 masks that overall include 298 agents of the initial set A . The network is consisted of 6 components, and 54 agents are not included in any mask. The 5 masks with the highest t_{CI} (the ones with the widest edges in Fig. 1b) include agents which participated in few projects, with behavioral profiles characterized by few changes over time. The reason is that these masks are identified as highly integrated as the activity of the agents involved is almost constant. Although low levels of entropy are generated, given that the activity of the involved agents is close to minimum, they can not be considered as the most relevant subsets. As these 5 not conducive masks generate independent components, the ongoing analyses are focused on the remaining 29 masks, which determine the largest component of 222 agents.

After the computation of the weighted betweenness centrality, the first results suggest a modification in the rank of centrality of the nodes. Although in the real-observed network, where agents are connected together if they co-participated in projects, the centrality of agents is related to the number of projects in which they participated, in the resulting network of RI masks this does not apply. More specifically, in the largest component of the one-mode projection of the weighted bipartite graph determined by the final set of masks $\mathcal{F}_{24,0.325,0.225}$, the following elements are emerging: (i) nodes with the largest number of participations in projects appear to be close to each other in one periphery of the network; (ii)

¹⁸ In case of agents belonging together to more than one community, the corresponding t_{CI} have been summed.

nodes with the smallest number of participations in projects appear to be close to each other in the opposite periphery of the network (with respect to the nodes with large number of projects); (iii) nodes with an average number of participations in projects appear to be very central; (iv) nodes with a high number of participations in projects and nodes with few participations in projects present few direct connections between them; (v) the shortest paths between very active nodes and scarcely active nodes (in terms of participations in projects) pass through agents with average activity.

The centrality ranking that can be inferred after these initial results reveals an entire change with respect to the observation in the original network of participation in projects. As the RI methodology allows the investigation of the joint integration of agents' dynamics, these first insights suggest that the agents with average number of activities, that now are the most central, harmonize the very intense activity of the nodes with many participations, namely the most central in the network of projects, with the scarce activity of those agents that participated in few projects. While the structure of the observed network of participations indicates that one of the most important and recognized laws of real complex network is respected, i.e. preferential attachment, the analysis of the functionality reveals insights that suggest new interpretations. These insights will be addressed in future research. Currently, because of the tests on the 447 considered partitions, observations do not suggest contradictory indications.

8 Conclusions

As physical order is addressed as a key dimension to the comprehension of the operation and the evolution of socio-economic complex systems [3], the main aim of this research is to contribute in the development the analysis of the entropy of joint behavioral time dynamics characterized by discontinuity, e.g. interactions. The objective of this work is to facilitate the implementation of a methodology that detects functional meso-structures with information theories [11–13]. In addition, the present work attempts to facilitate the implementation of entropy-related methods in the field of social sciences, and in particular in the analyses of socio-economic dynamic complex networks. The RI algorithm is extended with the PoSH-CADDy three-step methodology so as to reduce redundancy issues. The proposed approach is implemented in a real-world dynamic network (economic agents participating to Region Tuscany Network Policies from 2000–2006) consisted of ≈ 350 agents, where the proportion between the number of agents and the number of instants is $\approx 6:1$. In a complex dynamic network where the number of time instants is considerably lower than the number of involved agents, the proposed procedure accomplished to successfully detect a final set of 34 RI masks representing 34 groups of agents, whose behaviors are considered as integrated, namely not random. For the scope of this study, the focus is set in those partitions with a minimum percentage of agents included in at least one mask, and without too many overlaps among masks. The revealed ranking of the nodes' centrality appears to be substantially different from the one observed in the network of participations in projects.

In the perspective of this research are (i) the development of analytic models to statistically describe agents' characteristics in relation to the topology of the network of RI masks, (ii) the analysis of partitions obtained by combinations of the presented parameters of the methodology, (iii) the implementation of the methodology in other case studies, (iv) and the implementation of the methodology based on the edges' activation over time, instead of agents' statuses, as system variables.


References

1. Wasserman, S., Faust, K.: *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge (1994)
2. Righi, R., Roli, A., Russo, M., Serra, R., Villani, M.: New paths for the application of DCI in social sciences: theoretical issues regarding an empirical analysis. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016*. CCIS, vol. 708, pp. 42–52. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_4
3. Hidalgo, C.: *Why Information Grows: The Evolution of Order, from Atoms to Economies*. Basic Books, New York (2015)
4. Villani, M., Filisetti, A., Benedettini, S., Roli, A., Lane, D., Serra, R.: The detection of intermediate-level emergent structures and patterns. In: *ECAL*, pp. 372–378 (2013)
5. Villani, M., Benedettini, S., Roli, A., Lane, D., Poli, I., Serra, R.: Identifying emergent dynamical structures in network models. In: Bassis, S., Esposito, A., Morabito, F.C. (eds.) *Recent Advances of Neural Network Models and Applications*. SIST, vol. 26, pp. 3–13. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04129-2_1
6. Filisetti, A., Villani, M., Roli, A., Fiorucci, M., Serra, R.: Exploring the organisation of complex systems through the dynamical interactions among their relevant subsets. In: *Proceedings of the European Conference on Artificial Life 2015 (ECAL 2015)*, vol. 13, pp. 286–293 (2016)
7. Fortunato, S.: Community detection in graphs. *Phys. Rep.* **486**(3), 75–174 (2010)
8. Fortunato, S., Hric, D.: Community detection in networks: a user guide. *Phys. Rep.* **659**, 1–44 (2016)
9. Sani, L., et al.: Efficient search of relevant structures in complex systems. In: Adorni, G., Cagnoni, S., Gori, M., Maratea, M. (eds.) *AI*IA 2016*. LNCS (LNAI), vol. 10037, pp. 35–48. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49130-1_4
10. Vicari, E., et al.: GPU-based parallel search of relevant variable sets in complex systems. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016*. CCIS, vol. 708, pp. 14–25. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_2
11. Tononi, G., McIntosh, A.R., Russell, D.P., Edelman, G.M.: Functional clustering: identifying strongly interactive brain regions in neuroimaging data. *NeuroImage* **7**(2), 133–149 (1998). <https://doi.org/10.1006/nimg.1997.0313>
12. Tononi, G., Sporns, O., Edelman, G.M.: A measure for brain complexity: relating functional segregation and integration in the nervous system. *Proc. Natl. Acad. Sci. U.S.A.* **91**(11), 5033–5037 (1994). ISSN: 0027-8424
13. Tononi, G., Sporns, O., Edelman, G.M.: A complexity measure for selective matching of signals by the brain. *Proc. Natl. Acad. Sci. U.S.A.* **93**(8), 3422–3427 (1996). ISSN: 0027-8424

14. Russo, M., Rossi, F.: Cooperation networks and innovation: a complex system perspective to the analysis and evaluation of a EU regional innovation policy programme. *Evaluation* **15**, 75–100 (2009). <https://doi.org/10.1177/1356389008097872>
15. Caloffi, A., Rossi, F., Russo, M.: The emergence of intermediary organizations: a network-based approach to the design of innovation policies. In: *Handbook on Complexity and Public Policy*, pp. 314–331 (2015). ISBN: 978-1-78254-951-2
16. Rossi, F., Caloffi, A., Russo, M.: Networked by design: can policy requirements influence organisations' networking behaviour? *Technol. Forecast. Soc. Chang.* **105**, 203–214 (2016). <https://doi.org/10.1016/j.techfore.2016.01.004>
17. Lane, D.A.: Complexity and innovation dynamics. In: *Handbook on the Economic Complexity of Technological Change*. Edward Elgar Publishing (2011). ISBN: 978-0-85793-037-8



Modeling Urbanization Perception: Emerging Topics on Hangzhou Future Sci-Tech City Development

Debora Slanzi^{1,2} , Valentina Anzoise¹, and Irene Poli^{1,2}

¹ European Centre for Living Technology, S. Marco 2940, 30124 Venice, Italy
{debora.slanzi, valentina.anzoise, irenepoli}@unive.it

² Department of Environmental Sciences, Informatics and Statistics,
Ca' Foscari University of Venice, via Torino 155, 30172 Venice, Italy

Abstract. The complexity of the study on urban systems poses the challenging problem of developing methodological approaches for analyzing and modeling social data, both from a quantitative and a qualitative perspective. This work presents the research conducted to explore the perception on the urban development of an high-tech zone which has been recently established in China, i.e. Hangzhou Future Sci-Tech City. We conducted field research and collected data to identify which topics, concepts and interpretative categories are embedded in the social discourses about urban development and to derive the network of relations typical of complex social systems. The results of these analyses suggest that the perception of the people interviewed is mostly of great appreciation for the economic development with some concerns on the negative effects of this development on the society and the environment.

Keywords: Urbanization · Perception · Structural Topic Models
Network of relations

1 Introduction

Recent social and economic literature has been particularly concerned with urban development [1–3]. As it is well known, in China the urbanization rate and the urban population have had huge increases: in 1979, only the 18% of Chinese population lived in cities, but this surged to 54% by 2013 and the number of cities increased from 193 to 9000 (<http://mediumcities-china.org>). This process of accelerated urbanization is the result of rapid economic growth and dedicated policies: with more than 80% of global GDP generated in cities, urbanization in fact contributes to China's growth by increasing its productivity and allowing innovation and new ideas to emerge. However, as a consequence, cities and countrysides have changed at an unprecedented scale and pace: landscape and lifestyle have radically transformed raising social, economic, and environmental sustainability issues and stability problems [4, 5]. The changes and transformations generated by the planning of new urban areas have introduced new

imaginaries and new representations of livability and sustainability. These developments are differently perceived by the populations living or experiencing what can be considered a *rural to urban* transition without precedent in human history. The analysis of how citizens feel these transformations can therefore provide important elements to grasp at what people imagine and see about such a dramatic urban development. Due to the complexity of the study of urban systems, there is an increasing need of adopting appropriate methods for analyzing and modeling social data, both from a quantitative and a qualitative perspective [6]. In this work, we conducted an evolutionary research design where data are collected through a set of interviews to a selection of stakeholders of a recently established high-tech zone in China, i.e. Hangzhou Future Sci-Tech City. This area, 113 km² large, was previously covered by farmlands and it is now benefiting of dedicated-national policies to implement strategies to attract talents, improve scientific and technology innovation and foster new entrepreneurship. These new territorial entities are producing different consequences on the economy, on the environment and on the landscape, both positive and negative, which generate different perception on their development.

Aim of this work is to identify which topics, concepts and interpretative categories are embedded in the social discourses about Hangzhou and Future Sci-Tech City development and derive the corresponding network of relations typical of complex social systems. In particular, to let the main latent themes to emerge from citizens perception, we adopt narrative and textual data analysis approaches on the collected interviews. Texts in fact provide a valuable source of data for the identification and the measurement of latent variables, and several methodological approaches to study these structures of data have been proposed [7–9]. In particular, Topic Modeling approaches (TM) aim to infer from textual data the latent topics of sets of documents or texts [7]. These models have been successfully used across a variety of fields [10, 11]. A network of the relations among the estimated latent topics about the perception of this new area development is then derived by the analysis of the relevant factors emerged by TM analysis.

The paper is organized as follows: in Sect. 2 we illustrate the materials and methods of the study presenting the research design that we developed to produce the data and the textual data analysis approaches; then in Sect. 3 we present and discuss the research results on the perception and its modeling. Finally, in Sect. 4 we propose some concluding remarks about issues requiring further research.

2 Materials and Methods

2.1 The Perception on Hangzhou Future Sci-Tech City

In this work we analyze the perception on the urban development of a recently established high-tech zone in China, i.e. Hangzhou Future Sci-Tech City. At this aim we conducted several in depth interviews in the period Spring-Summer 2016 to different type of stakeholders, ranging from residents and workers to planners



Fig. 1. Selection of 32 photos used to conduct the interviews.

and students. Moreover to elicit the perception of the different stakeholders, the interviews were conducted using a composition of 32 photos of the area which each stakeholder could comment and select in order to describe his/her narrative about the changes occurred in the Future Sci-Tech City area. Images are inherently polysemic, but each of those used within this research poses the focus on different aspects of the development of the area, such as environmental, socio-cultural, economic and working condition aspects [12]. The selection of photos used in the interviews is presented in Fig. 1.

We then identify the core elements of the stakeholders' narrative by means of Nvivo, a computer assisted qualitative data analysis software: text coding has been conducted achieving the identification of nodes (labels) for the main conceptual categories. Then short texts corresponding to specific labels, here identified by the 32 photos, have been extracted and elaborated to estimate a statistical model for textual analysis. In this way we have been able to focus on the specific perceptions highlighted and elicited by the proposed images and

Table 1. Examples of short texts used in the analysis.

Photo	Short text
P21	I am sure that most of people would say “Oh, how beautiful are those new buildings”, like these [P3, P16, P21, P26] and they will not like this [P23]
P22	Many local people are living here [P22] and the environment is not very good
P27	There is too much rubbish and waste [P27, P14]. Perhaps, the people of our country do not pay much attention to this aspect. We only think how to build beautiful houses, but in terms of ecological environment, we do not care

on the latent issues they let emerge. Examples of short texts are presented in Table 1. From this table we can see that each photo is often commented together with other photos. Then latent topics emerging from the analysis of these coded texts can reveal relational structures that can be derived from the analysis of photo comments and narratives. Therefore, the research design developed in this work is composed by the following steps:

- Estimate latent topics of urbanization perception from the short texts related to photos;
- Build the graph of co-citation among photos, which means to identify the connections emerged when two or more photos are commented together;
- Derive the network of topic relations by merging the information extracted from topic contents and the photo co-citation graph.

Each design procedure will be briefly described in details in the following sections.

2.2 Structural Topic Models

Topics are estimated with probabilistic distributions over a vocabulary of words and according to the co-occurrence of words within each analyzed text with a probabilistic generative process. This process considers a collection of D documents (or texts), each containing $N_d \subseteq V$ words, $d = 1, \dots, D$, and V represents the set of distinct elements (words) of the vocabulary used in the analysis. Moreover a set of K latent topics is defined and assumed to be representative of the documents. The probabilistic generative process consists then of the following steps:

- a V -dimensional Dirichlet probability distribution, $\beta_k \sim Dir(\eta)$, is determined for each topic k , $k = 1, \dots, K$, assessing the probability according to which words are generated from the k -th topic;
- a K -dimensional Dirichlet probability distribution, $\theta_d \sim Dir(\alpha)$ is determined for each document d , $d = 1, \dots, D$, assessing the expected proportion of words that can be attributed to each topic;

- for each word in the document
 - a value $z_{d,n}$ for a multinomial distribution $Z_{d,n} \sim Mult_K(\theta_d)$, $n = 1, \dots, N_d$, is sampled denoting which topic is associated with such word, and
 - a word value $w_{d,n}$ from a multinomial distribution $W_{d,n} \sim Mult_V(\mathbf{B}z_{d,n})$, is sampled where the matrix $\mathbf{B} = [\beta_1 \dots \beta_K]$ encodes the distributions over words in the vocabulary associated with the K topics.

When additional information regarding the documents is available, it can be included in the model as a set of covariates \mathbf{X} . The Structural Topic Model (STM) proposed by Roberts et al. [13–15] represents a particular class of TMs where the inclusion of covariates of interest can affect the topical prevalence (i.e., the frequency with which a topic is discussed) and the topical content (i.e., the distribution with which the words are used to discuss a topic) of the model. The covariates are introduced in the TM approach by means of different prior probability distributions for document-topic proportions and topic-word distributions. In this research, we consider the photos as additional information with respect to interviews’ texts to highlight if particular visual stimuli bring out specific perceptions of latent issues. For the specific procedure on how these prior distributions are defined and how the TM estimation process is modified, we refer to [15]. Several pre-processing analyses are conducted to remove irrelevant words and symbols in the texts as well as to stem (reduce words to their root form). Not frequent words, i.e. words which are present only in the 1% of the analyzed documents, are also removed. The resulting dataset is composed by $D = 319$ texts on $|V| = 384$ words and a covariate $X = x_1, x_2, \dots, x_{32}$ indicating the photo from which the text is extracted as indicated in Table 1. A search across models with different number of topics was also performed to identify the preferable number of topics for the model estimation; the achieved number is equal to $K = 10$.

The analyses were conducted using the package `stm` of the R-project free software environment for statistical computing and graphics (www.r-project.org) [16].

2.3 Network Analysis

From the STM results we can build the *graph of photo co-citation*. Photos, as covariates of the models, have associated their particular vocabulary of words from which we can derive which words of a photo are part of the set of words of other photos. In this way, we can build a graph of co-citation by linking photos which have a directed connections with others. In particular, we build the photo co-citation graph in the following way:

- if a photo x_j is part of the covariate vocabulary of the photo x_i , $i, j = 1, \dots, 32$ and $i \neq j$, then a link between x_i and x_j is derived;
- links are bidirectional if the covariate vocabulary of a photo includes photos which have in their vocabulary the citation to the generating photo itself.

This graph is then used to empirically derive the network of topic relations by merging its information with the information extracted from topic contents. From the STM results we can in fact detect that several photos are particularly related to specific topics and the links between photos produce a flux of information which can give rise to the relationships among topics. The network of topic relation is represented as an undirected graph, where nodes correspond to the estimated topics and edges between nodes are built considering that if a cluster or a sub-cluster of photos (i.e. a group of connected components) is related to more than one topic, then the corresponding topics are connected and an edge between the nodes of the graph is drawn.

3 Results

The latent topics estimated by means of STM are presented in Table 2. In this Table, topics are described by their most frequent/exclusive words identified by FREX measure [15], a metric which combines word frequency and exclusivity, and by the estimate value of θ , the expected proportion of words that can be attributed to each topic. Moreover as topic words can be represented also by photos (photos are in fact part of the short text vocabulary as shown in Table 1), we report also the most frequent and exclusive photo numbers for each estimated topic. Here, topics are labeled by the most representative theme which they express and are ranked based on their estimated expected proportion. We can see that most of the topics have the same proportion across the documents, i.e. Topic 2, 3, 4, 5, 6, and 7 are equally distributed along the texts, meaning that they are discussed in the same extent during the interviews. There is one topic with higher expected proportion, Topic 1, highlighting that the theme of Landscape quality is discussed more frequently than others, whereas Topic 8, 9 and 10 are the less present in the texts.

Then, following the procedure introduced in Sect. 2.3, we can derive the co-citation graph among photos investigating the resulting STM vocabulary of words of each photo. This graph is reported in Fig. 2. From this graph, we can see that there are some “clusters” of photos (see for example, P4 + P17 + P22 or P19 + P22 + P25 + P28) meaning that in the interviews people citing a photo have high probability of citing other photos which are connected as reported in the graph. We can also notice different clusters of photos with respect to the structure of the connections: some photos are not connected (for example, P2 or P12), some photos are connected in unidirectional way (P25 \rightarrow P9) and some others in multi-directional way (P25 \leftrightarrow P22).

The information extracted from the co-citation graph is then merged with the achieved topic contents the derive the network of topic relation. The resulting network is shown in Fig. 3.

From this graph, several measures are calculated to identify a set of characteristics of the estimated topics. We derive:

- the degree of each node, i.e. the number of its adjacent edges. Usually, high values of degree indicates how the node is connected to other nodes: in the

analysis of the urban development perception, the measure can indicate the extent to which a topic is able to produce a narrative flux which includes or makes emerge other themes perceived as relevant.

- the standardized betweenness measure of centrality, i.e the standardized number of the shortest paths that pass through the node. It represents the degree of which nodes stand between each other, and higher betweenness centrality indicated that more information will pass through that node. In this analysis it can indicate how a topic is central in the a flux of discourses on the perception.
- the normalized eigenvector centrality scores, i.e. the eigenvector corresponding to the largest eigenvalue of the graph adjacency matrix. The scores are normalized so that the sum of all scores is 1. This is a measure of the influence of a node in the graph, and it can therefore highlight how a topic is able to influence the emergence of other topics in the interviews.

These measures are presented in Table 3. They are usually correlated each other, with some little distinctions. However, they provide information about which topics are perceived and discussed as the most relevant by those stakeholders who are differently affected and involved in the urban development of this area of Hangzhou. In particular we see that Topic 3, i.e. Progress and modernity, seems the key theme of the network of topic relations despite its expected mean

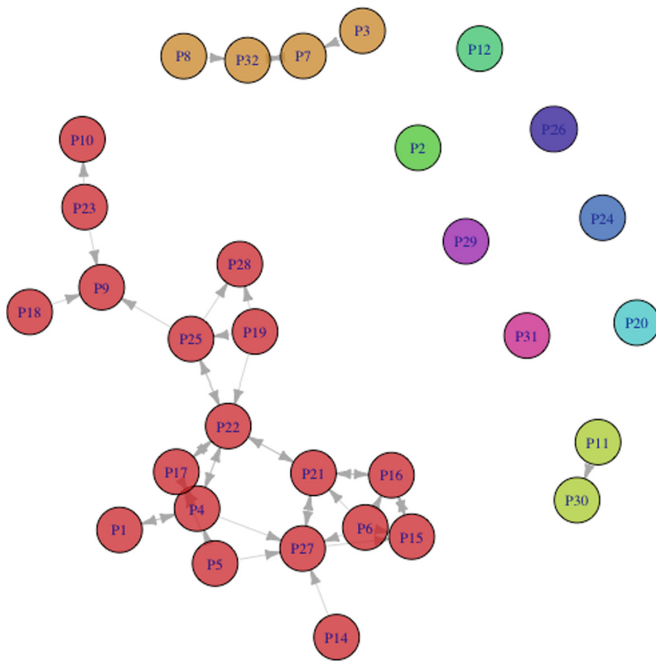


Fig. 2. Co-citation graph of photos built by evaluating the words identified for each photo in the estimated model

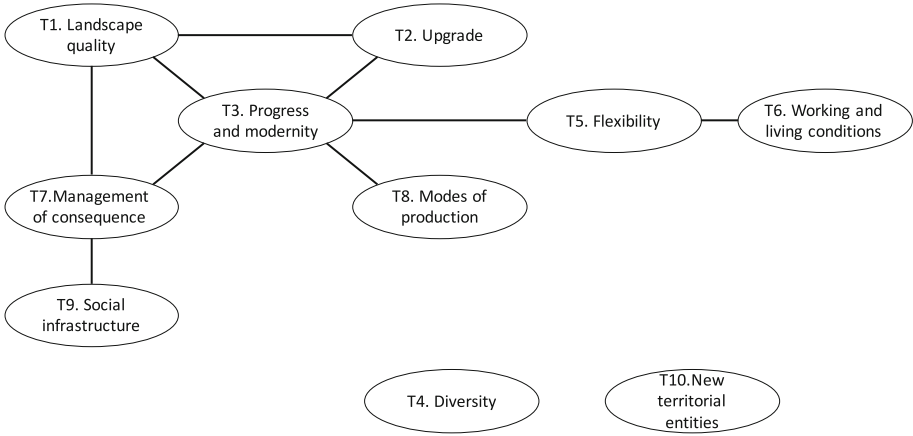


Fig. 3. Network of topic relations.

Table 2. Most frequent and exclusive words, including photos, and expected mean proportion of each topic.

Topic	Most frequent and exclusive words (and photos)	Expected proportion
T1. Landscape quality	Place, peac, see, beauti, local, gone, new, countrysid, acient, demolish (23, 16, 21, 10)	15.1%
T2. Upgrade	Look, care, good, better, area, live, construct, subway, presenrv, origin (3, 15, 7, 16, 21)	12.2%
T3. Progress and modernity	Futur, sci-tech, west, shop, road, style, develop, time, past, progress (27, 21, 22, 15, 16)	11.8%
T4. Diversity	Care, farmer, compani, cultur, design, want, need, shanghai, internet, concern (5, 3, 10)	11.4%
T5. Flexibility	Develop, maintain, keep, disappear, flexibi, help, hope (17, 4, 22, 21, 1)	11.3%
T6. Working and living conditions	Chang, recogn, differ, work, lif, low, direct, feel, compens, fake (22, 28, 12, 4, 24, 27, 1)	11.1%

(continued)

Table 2. (*continued*)

Topic	Most frequent and exclusive words (and photos)	Expected proportion
T7. Management of consequences	Good, look, recogn, haichuangyuan wetland, environmen, build, govern, suitabl, order, worry, temporary (5, 15, 25, 32)	10.8%
T8. Modes of production	Comparison, cold, disord, connect, rubbish, eat, dirti, labour, indic (7, 22, 4, 21)	7.1%
T9. Social infrastructures	Street, people, simpli, talk, familiar, house, village, rememb (4, 27, 1, 21, 24)	6.2%
T10. New territorial entities	Center, custom, land, pollut, area, miss, resid, space, work, canal, rich, buy (21, 15)	3%

Table 3. Measures of centrality of the estimated topics.

	Degree	Betweenness	Eigenvector
T1	3	0.022	0.190
T2	2	0.000	0.147
T3	5	0.333	0.234
T4	0	0.000	0.000
T5	2	0.133	0.092
T6	1	0.000	0.032
T7	3	0.133	0.167
T8	1	0.000	0.081
T9	1	0.000	0.058
T10	0	0.000	0.000

proportion is not the highest. This topic mainly concerns with people's attributions about modernization and China's "moving forward" process, in which the agents, purposes and consequences entailed by the development of high-tech zone seems to play a crucial role. The imperative of modernization seems therefore the central narrative that is embedded in people's perceptions, and discourses about Landscape quality (Topic 1) and the Management of consequences (Topic 7) necessarily relate to it. Topic 1 concerns the distinguishing attributes of the natural

and social ecosystems characterizing the Future Sci-Tech City, and also how different factors and features interplaying one each other, or different decisions and behaviors, can influence their status and evolution, whereas Topic 7 concerns handling the positive and negative impacts of the Sci-Tech City development. In particular interviewees refer mainly to how this will affect its economy, heritage, natural resources, communities' life, health, etc. From these results we can see that the majority of interviewed stakeholders are aware of the dramatic consequences of such a rapid transition but they expect that - sooner or later - the negative impacts will be properly managed. At the same time the current imperative of modernization and material wellbeing seems to be the main priority in order to achieve the so called "Chinese Dream" which claims ambitious targets for 2020, in particular sustaining the growth of population living in cities and expanding domestic consumption at a "sustainable pace".

4 Concluding Remarks

Perceptions on the urbanization of Hangzhou fostered by the planning of new urban areas such as the Future Sci Tech City have been collected with depth interviews based on photos and narratives. The textual documents resulting from the interviews have been analyzed and Structural Topic Models have been constructed and estimated. From these models we could identify the most relevant topics, as perceived by the stakeholders of the area, the relations among the topics and the structure of the network that characterizes these relations. A key result of this analysis is the extremely positive perception of the impressive economic growth experienced by this area since the establishment of the high-tech zone in 2011, and achieved also with the rapid population growth and urbanization. A tolerant attitude is shown instead towards emerging environmental and social issues, which are part of the compromises required by development and, therefore, are regarded as not as urgent to address in this transition phase.

Acknowledgements. This work was supported by the EU-CHINA Research and Innovation Partnership, EuropeAid/135-587/DD/ACT/Multi EU Project: New pathways for sustainable urban development in China's medium-sized cities (MEDIUM). This publication has received funding from the European Union under the External actions of the European Union - Grant Contract ICI+/2014/348-005. The contents of this publication are the sole responsibility of the authors and can in no way be taken to reflect the views of the European Union.

References

1. Zhao, P., Li, P.: Rethinking the relationship between urban development, local health and global sustainability. *Curr. Opin. Environ. Sustain.* **25**, 14–19 (2017)
2. Yang, B., Xu, T., Shi, L.: Analysis on sustainable urban development levels and trends in China's cities. *J. Clean. Prod.* **141**, 868–880 (2017)
3. Riffat, S., Powell, R., Aydin, D.: Future cities and environmental sustainability. *Future Cities and Environ.* **2**, 1 (2016)

4. Wu, F.: Emerging Chinese cities: implications for global urban studies. *Prof. Geogr.* **68**(2), 338–348 (2016)
5. Wei, Y.H.D.: Restructuring for growth in urban China: transitional institutions, urban development, and spatial transformation. *Habitat Int.* **36**, 396–405 (2012)
6. Dolfin, M., Leonida, L., Outada, N.: Modeling human behavior in economics and social science. *Phys. Life Rev.* (2017, in press)
7. Blei, D.M.: Probabilistic topic models. *Commun. ACM* **55**(4), 77–84 (2012)
8. Grimmer, J., Stewart, B.: Text as data: the promise and pitfalls of automatic content analysis methods for political documents. *Poli. Anal.* **21**(3), 267–297 (2013)
9. Li, G., Feng, S., Jun, T.: Textual analysis and machine learning: crack unstructured data in finance and accounting. *J. Financ. Data Sci.* **2**(3), 153–170 (2016)
10. Tvinnereim, E., Liu, X., Jamelske, E.M.: Public perceptions of air pollution and climate change: different manifestations, similar causes, and concerns. *Clim. Change* **140**, 1–14 (2016)
11. Reich, J., Tingley, D., Leder-Luis, J., Roberts, M.E., Stewart, B.M.: Computer-assisted reading and discovery for student generated text in massive open online courses. *J. Learn. Anal.* **2**(1), 156–184 (2015)
12. Anzoise, V.: Perception and (re)framing of urban environments: a methodological reflection toward sentient research. *Vis. Anthropol.* **30**(3), 177–190 (2017)
13. Roberts, M.E., Stewart, B.M., Tingley, D., Airoidi, E.M.: The structural topic model and applied social science. In: *Neural Information Processing Society* (2013)
14. Roberts, M.E., Stewart, B.M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S.K., Albertson, B., Rand, D.G.: Structural topic models for open-ended survey responses. *Am. J. Polit. Sci.* **58**(4), 1064–1082 (2014)
15. Roberts, M.E., Stewart, B.M., Airoidi, E.M.: A model of text for experimentation in the social sciences. *J. Am. Stat. Assoc.* **111**(515), 988–1003 (2016)
16. Roberts, M.E., Stewart, B.M., Tingley, D.: STM: R-package for structural topic models. R package version 1.2.2 (2013). <http://www.structuraltopicmodel.com>

Complexity



Complexity Measures in Automatic Design of Robot Swarms: An Exploratory Study

Andrea Roli¹, Antoine Ligtot², and Mauro Birattari²

¹ Department of Computer Science and Engineering, Campus of Cesena, Alma Mater Studiorum, Università di Bologna, Cesena, Italy
andrea.roli@unibo.it

² IRIDIA, Université libre de Bruxelles, Brussels, Belgium

Abstract. The design of control software for robot swarms is a challenging endeavour as swarm behaviour is the outcome of the entangled interplay between the dynamics of the individual robots and the interactions among them. Automatic design techniques are a promising alternative to classic *ad-hoc* design procedures and are especially suited to deal with the inherent complexity of swarm behaviours. In an automatic method, the design problem is cast into an optimisation problem: the solution space comprises instances of control software and an optimisation algorithm is applied to tune the free parameters of the architecture. Recently, some information theory and complexity theory measures have been proposed for the analysis of the behaviour of single autonomous agents; a similar approach may be fruitfully applied also to swarms of robots. In this work, we present a preliminary study on the applicability of complexity measures to robot swarm dynamics. The aim of this investigation is to compare and analyse prominent complexity measures when applied to data collected during the time evolution of a robot swarm, performing a simple stationary task. Although preliminary, the results of this study enable us to state that the complexity measures we used are able to capture relevant features of robot swarm dynamics and to identify typical patterns in swarm behaviour.

1 Introduction

The behaviour of a swarm of robots is the result of the dynamic interplay among the robots, and between robots and environment. As a consequence, the design of control software for a robot swarm presents hard challenges. Typical techniques for designing robot swarm are based on code-and-fix methods [4], usually tailored to the specific problem at hand. A promising alternative to these *ad hoc* approaches is provided by automatic design techniques [9], which are especially suited to deal with the inherent complexity of swarm behaviours. In automatic methods, the design problem is cast into an optimisation problem, whereby the solution space contains instances of control software and an optimisation algorithm is applied to tune the free parameters of the architecture [10, 29]. For the

sake of completeness, we observe that the design process is not simply reduced to an optimisation problem because it also involves the definition of proper merit factors and experimental settings, likewise learning methods.

Recently, some information-theoretical measures have been proposed for the analysis and design of the behaviour of single autonomous agents [1, 8, 25, 28]. These studies support the use of information theory and complexity science concepts in the field of autonomous agents and robotics. We believe that these techniques may be fruitfully applied also to swarms of robots. Indeed, complex systems science may provide a corpus of theories and methods that enable the designer to formally and quantitatively analyse the dynamics of a robot swarm and its internal information processes.

Complexity measures may be applied to the automatic design of robot swarms with the following objectives:

1. understanding individual and swarm behaviour from observations of measurable quantities (e.g. sensor readings, actuation, controller state);
2. providing task-agnostic merit factors for the automatic design procedures;
3. classifying swarm tasks in terms of their intrinsic complexity so as to optimally tune the complexity of individual robot and robot interactions.¹

In the long term we plan to address the following questions: *(i)* Are the intuition behind the measures in accordance with the observed robot swarm behaviour? And is the observed behaviour coherent with the complexity values measured? *(ii)* What are the most informative measures? *(iii)* What are the complexity measures most suited for such an application? *(iv)* Are there phenomena in the swarm behaviour that can be detected just by observing the complexity values measured? The outcome of this study is expected to provide guidelines for the choice of the most informative indicators for more complex tasks.

In this work, we present a preliminary study on the applicability of complexity measures to robot swarm dynamics. The aim of this investigation is to compare and analyse prominent complexity measures when applied to data collected during the time evolution of a robot swarm performing a simple stationary task. In the following, we first summarise the measures considered in this study in Sect. 2; subsequently, we detail the robot swarm task (Sect. 3). In Sect. 4, we provide a summary of the main results, emphasising the ones that enable us contributing to answer the questions raised above. Finally, we conclude with an outlook of ongoing and future work.

2 Measures of Complexity

In the scientific literature the word *complexity* is overloaded, as it appears with different meanings, each related to a specific interpretation of the term. As a consequence, there is no unique measure of complexity and in fact many metrics have been proposed in the literature. In general, each metric addresses one

¹ This goal is motivated by a conjecture on the so-called reality gap, which has been advanced in [3, 10].

specific aspect of the general notion of complexity, therefore we should aim at producing a *complexity fingerprint* by evaluating several measures, rather than identifying a single metric able to summarise all the relevant properties related to complexity.

A measure of complexity was first proposed by Kolmogorov [14] who provided an algorithmic view of information: the complexity of a string of symbols is defined as the length of the shortest program producing it. As this measure is not computable some approximations have been proposed, such as the ones based on compression algorithms. In fact, algorithmic complexity estimates the amount of randomness in a string, as they turn out to be very low for regular sequences and maximised for completely random strings. The definition of complexity we are interested in tries to capture the notion of presence of (dynamical) patterns, often related to the extent to which correlations distribute across the element of the system observed [13]. Intuitively, high complexity is associated to situations between order and disorder, as patterns in both ordered and completely random dynamics are negligible. Along this line, several measures have been proposed [12, 13, 16, 18, 26]. However, a survey on the literature on complexity measures is out of the scope of this contribution and we refer the interested reader to prominent works on the subject [2, 13, 18, 20, 24]. A nice introduction to information theory for complex systems can be found in the lecture notes by Lindgren [17].

In this work, we focus primarily on the complexity of the dynamics of the system observed in its environment, rather than the individual complexity of a controller of an isolated robot. Moreover, as a consequence of the fact that we deal with data collected during experiments, the measures used should be applied to time series of finite length. Among the measures proposed in the literature, we selected and implemented the following ones:

1. Shannon entropy [27]
2. Block entropy and entropy excess [22]
3. Correlation information [17]
4. Mutual information [6]
5. LMC complexity [21]
6. Lempel-Ziv complexity [15]
7. *bz2* compression factor [5]
8. Linguistic complexity [30]
9. Set-based complexity [11]

The choice of these metrics has been motivated by the intent of covering the diverse facets of complexity, and also taking into account computational requirements.²

Measures 1–5 are based on the Shannon entropy of a sequence s of symbols in a finite set \mathcal{X} . We suppose that the frequency of symbols appearing in s approximates the probability distributions of the symbols. Therefore, we can

² Indeed, due to excessive computational resources required, for this preliminary step we did not applied measures of complexity based on model construction, such as the ones by Crutchfield et al. [7].

provide the definition of entropy in terms of random variables. Let X be a random variable which can assume values from a finite and discrete domain \mathcal{X} , the Shannon entropy of X is defined as

$$H(X) = - \sum_{x \in \mathcal{X}} P(x) \log P(x)$$

where the logarithm is expressed in base 2. This definition can be extended to blocks of symbols of length n in s , so as to take into account also correlations among symbols. This leads to the definition of the *block entropy* of length n :

$$H_n = - \sum_{s_n \in \mathcal{S}} P(s_n) \log P(s_n)$$

The *entropy excess*³ is defined as the difference between block entropies of length n and $n - 1$ and estimates the information required to predict the n -th symbol conditioned to the observation of $n - 1$ preceding symbols. In formulas:

$$h_n = \Delta H_n = H_n - H_{n-1}$$

We can extend this process to the second derivative (in discrete domains) and obtain the *correlation information* from length n :

$$k_n = \Delta^2 H(n) = -H(n) + 2H(n - 1) - H(n - 2), \quad n \geq 3$$

Intuitively, the peaks of k_n identify significant block regularities, i.e. maximum gain in information for specific block lengths.

Also the *mutual information* $I(X, Y)$ between random variables X and Y is defined in terms of entropies and estimates the average information one gains about Y after the observation of X , and viceversa:

$$I(X, Y) = H(X) + H(Y) - H(X, Y)$$

where $H(X, Y)$ denotes the conjunct entropy of X and Y .

For completeness, we also introduce the LMC complexity⁴ which is defined in terms of entropy and disequilibrium:

$$LMC(X) = H(X) \cdot D(X)$$

where $D(X) = \sum_{x \in \mathcal{X}} \left(P(x) - \frac{1}{|\mathcal{X}|} \right)^2$. Unfortunately, this metric is quite sensitive to numeric factors—mainly the values of H and D at the borders—and the results it returns should be taken with care.

Measures 6–9 are instead based on computing properties of the sequence at hand, rather than referring to a probability distribution. In particular, the

³ Not to be confused with the *excess entropy* [26], which is defined for $n \rightarrow \infty$.

⁴ The name comes from the name initials of its inventors.

Lempel-Ziv complexity (LZ) is a sort of algorithmic information measure computable on finite sequences, therefore it estimates the randomness of a string. $LZ(s)$ returns the number of shortest different blocks composing s . Along the same line is the compression factor achieved when compressing the string, in our case with algorithm *bzip*, which takes into account blocks of different size. *Linguistic complexity* is another metric that based on the occurrences of different blocks in a sequence of symbols and is computed for blocks of varying size.

Finally, the complexity of a set of strings $S = \{s_1, s_2, \dots, s_N\}$ can be estimated by means of the *set-based complexity SBC*, which accounts for the informative contribution of each string to the set. The intuition behind this measure is that a random string and a duplicated string do not contribute to the overall complexity of the set. This metric is defined in terms of Kolmogorov complexity $K(s)$ and it is empirically computed by approximating it with a compression algorithm, providing an estimation $\hat{K}(s)$. Based on algorithmic complexity, the *distance* between two strings can be computed as follows:

$$d(i, j) = \frac{\hat{K}(x \oplus y) - \min(\hat{K}(x), \hat{K}(y))}{\max(\hat{K}(x), \hat{K}(y))}$$

where $x \oplus y$ denotes the concatenation of strings x and y . The SBC of the set of strings S is defined as:

$$SBC(S) = \sum_{i=1}^N \hat{K}(s_i) F_i(S)$$

where

$$F_i(S) = \frac{2}{N(N-1)} \sum_{j \in \{1, 2, \dots, N\}, i \neq j} d_{ij}(1 - d_{ij}), \quad d_{ij} := d(s_i, s_j)$$

3 Case Study: Random Walk with Collision Avoidance

We defined a case study that requires a simple software controller for the robots and few parameters to be tuned. Moreover, the mission the swarm has to accomplish should be modelled as a stationary process, and its level of complexity should be sufficiently high to be measured and produce non-trivial results. At the same time, the complexity should be limited so as to allow an easy interpretation of the results. We performed our experiments in a simulated environment by the means of ARGoS [23], one of the most widespread swarm robotics simulators. The robot chosen to be simulated is an *e-puck*, equipped with 8 infra-red proximity sensors positioned around the circular body and two wheels.

3.1 Behaviour: Random Walk with Collision Avoidance

The random walk behaviour is a strategy for space exploration commonly used in swarm robotics. We implemented this strategy as the alternate execution of


```

Function ControlStep()
  if State == STRAIGHT then
    if StraighthSteps == 0 OR obstacle in front then
      State = LEFT or RIGHT with same probabilities;
      Bernouilli(0,5) == 1 ? State ← LEFT : State ← RIGHT;
      TurningSteps ← Uniform(0,  $\frac{R_a}{10}$ );
    else
      GoStraight();
      StraightSteps ← StraightSteps - 1;
    end
  end
  if State != STRAIGHT then
    if TurningSteps > 0 then
      if State == RIGHT then
        TurnRight();
      else
        TurnLeft();
      end
      TurningSteps ← TurningSteps - 1;
    else
      State ← STRAIGHT;
      StraightSteps ←  $W_s$ ;
      ControlStep();
    end
  end

```

Algorithm 1. Control step of the random walk behaviour executed every 100 milliseconds. The methods GoStraight(), TurnRight() and TurnLeft() are responsible for affecting the required values to the wheels actuator in order for the robot to move forward, rotate clockwise or anti-clockwise, respectively. The recursive call to ControlStep() allows the robot to verify the absence of obstacle before starting to move forward.

straight movements and static rotations: at each time step of an experiment, the e-puck robots can either move forward for a given distance or rotate at a given angle. In our implementation of the random walk behaviour the robots walk straight for a maximal distance W_s . After this maximal distance is travelled, or if an obstacle is perceived in front of the robot, the static rotation phase is triggered. During the rotation, a robot turns left or right with same probability, with a rotation angle given by a multiple of 10° taken uniformly between 0 and a maximal angle R_a . Once the robot has completed the rotation, it can once again move forward under the condition that no obstacles are on the way. Conversely, if the path is not clear in front of the robot, another static rotation phase is immediately started. Algorithm 1 resumes the behaviour that we implemented.

3.2 Experimental Settings

For this study, we executed multiple runs of the random walk behaviour with two parameters: the maximal straight distance $W_s \in \{10, 20, 30\}$ expressed in centimeters, and the maximal angle of rotation $R_a \in \{40, 90, 180\}$ expressed in degrees.

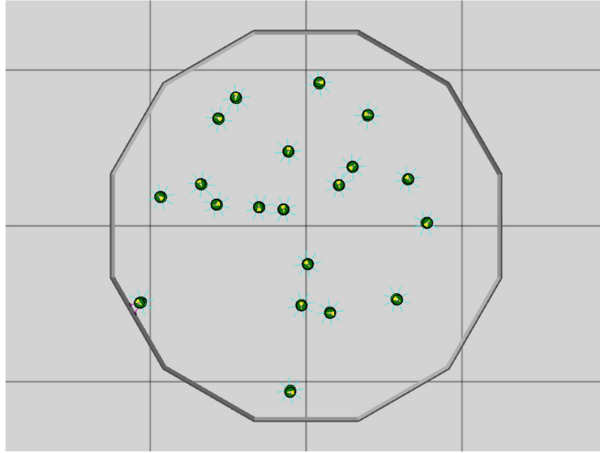


Fig. 1. Picture of the enclosed environment setup containing a swarm of 20 e-puck robots. The 8 cyan lines around each robots represent their proximity sensors. (Color figure online)

We ran two types of experiments. The first one is a control scenario involving a single robot that moves in an infinite space with no obstacles nor boundaries. This scenario represents a baseline for the comparisons with the swarms. The second experiment involves a swarm of $N \in \{1, 10, 20, 40\}$ robots moving in an enclosed environment in which the walls form a dodecagonal shape with an area equal to 4.91 m^2 (see Fig. 1). The swarm is composed of robots all controlled by the same random walk behaviour. At the beginning of each experiment, the robots are uniformly distributed in the dodecagonal arena. Every possible combinations of the parameters W_s and R_a were used in the two experiments. Each experiment was repeated 10 times. Therefore, a total of 450 experiments were ran.

The state of a robot performing this kind of random walk can be simplified and expressed by means of three possible states: Straight, Left, and Right. Hence, at each instant, the state of the whole swarm of N robots can be represented by a vector of symbols, each from the alphabet $\{S, L, R\}$. For each run, we recorded the state vector of the swarm every 100 ms. As runs last 20 min, a total of 12000 state vectors were recorded for each experiment. The complexity measures were applied to this symbolic sequence depending on the definition of the measure, i.e.,

either to the whole vector state (e.g., for set-based complexity) or by averaging the values computed across all the robots (e.g., for entropies).

4 Results

The factors influencing swarm behaviour that we expect to be reflected into a complexity metrics analysis are the number of forward steps, the maximum turning angle and the number of robots in the arena. In particular, the metrics should provide information on the amount of regularity in robots' trajectories and on their interactions. As we will show, although preliminary, the results of this analysis enable us to state that the complexity measures we used are able to capture these relevant features of robot swarm dynamics. Moreover, we discovered that some metrics were able to capture non-trivial properties of the dynamics of the swarm. In this paper we show and discuss a representative sample of the results. The metrics we have omitted in this discussion are anyway in agreement with the ones we have chosen for this presentation.

In the following plots, colours are used to differentiate among the three possible turning angle values: 40° in red, 90° in green and 180° in blue. The plots shown are produced by analysing one run for each possible combination of experimental factors; qualitatively analogous results are observed in the other runs.

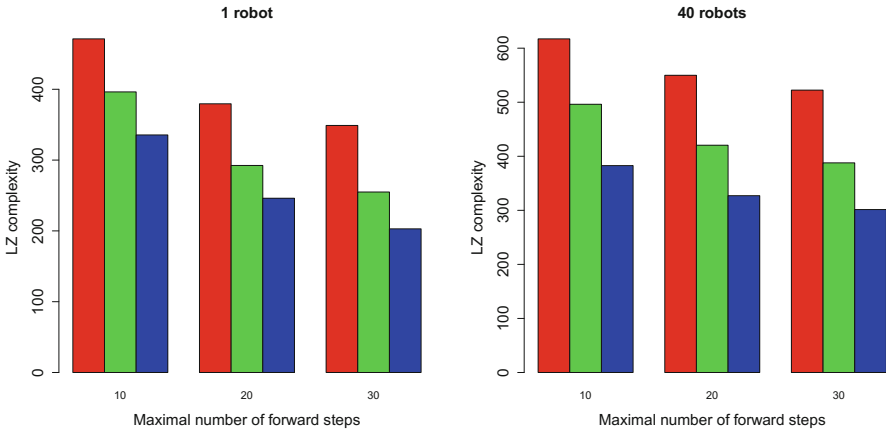


Fig. 2. LZ complexity for the case with 1 robot (left) and 40 robots (right) in the dodecagonal arena. Maximum turning angles: ■ 40° ■ 90° ■ 180° (Color figure online)

In general, Shannon entropy and all the metrics measuring randomness are in agreement with the expectations, as they show that randomness increases if the number of forward steps decreases, the maximum turning angle decreases or the number of robot increases. In Fig. 2 a representative example is shown for the LZ complexity. Note that the maximum values reached in the case of

40 robots are higher than those for one robot, providing a quantitative account of the positive correlation between number of robots and randomness in their trajectories. In addition, the LZ complexity decreases with the number of steps and the maximum turning angle, specifically confirming that robots' trajectories are more regular when they have more possibilities to avoid obstacles.

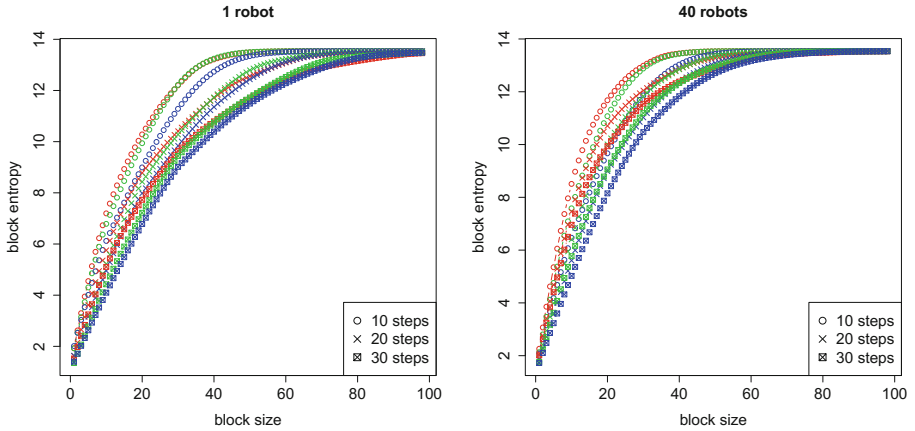


Fig. 3. Block entropy for the dodecagonal arena case, with 1 robot (left) and the 40 robots (right). Maximum turning angles: ■ 40° ■ 90° ■ 180°

Block entropies and their derivatives are particularly informative because they provide a picture of the correlations at different lengths in the dynamics of the swarm. The block entropy as a function of block size is plotted in Fig. 3 for the two extreme cases of the scenario with the dodecagonal arena. As expected, the curves grow more rapidly for the dynamics characterised by a higher level of randomness. The curves saturate when the length of the block considered is about 40; in fact, as data series are of finite length, the frequency of large blocks is underestimated and the block entropy values tend to converge even if, in principle, the asymptote should have a strictly positive derivative for non-periodic dynamics [17]. Therefore, the block entropy values are meaningful for shorter block lengths. The block entropy trends suggest two main observations. First, the initial slope of the curves is higher on average in the 40 robots case; this is a direct consequence of the fact that the denser the robots the less regular their trajectories in the arena. Second, the top and bottom limiting curves correspond to the least (10 steps, 40°) and most (30 steps, 180°) regular case, respectively.

The correlation information (i.e. the second discrete derivative of the block entropy) makes it possible to identify the points at which the block entropy slope changes, thus providing a tool for a detailed inspection of the regularities in the time series. The plots in Fig. 4 summarise the results of the correlation length analysis. The most notable fact to observe is that in every condition considered, and independently of the turning angle, there is a marked peak corresponding to

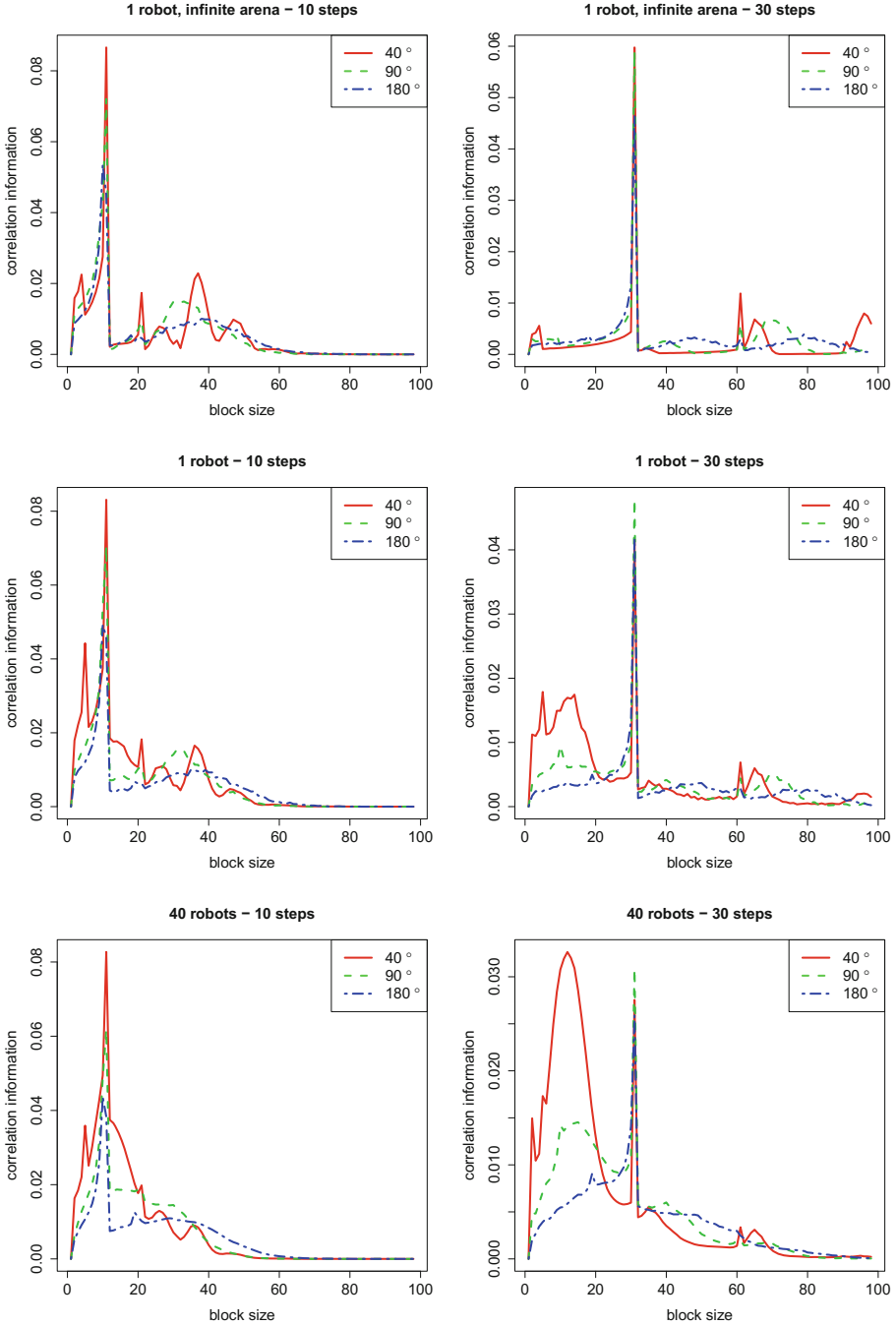


Fig. 4. Correlation information from length k , where k is the block length.

the number of forward steps. Indeed, this is one of the most relevant regularities in robots' trajectories. We can also observe lower peaks corresponding to multiples of the number of forward steps. This picture is particularly striking in the control case (one robot, infinite arena) and gets blurred when delimiting walls are present and mainly when robots in the arena are dense, as their avoidance behaviour introduces randomness in their trajectories. We observe also a surprising phenomenon: a second peak appears at the left of the previously mentioned one. This peak is particularly marked in the case of 40 robots and 30 forward steps, where it is even higher than the other peak. This second local maximum captures the pattern of turning moves of the robots trying to avoid an obstacle. Indeed, the location of this peak gives us an indication of the average number of turning moves the robots have to take before finding a free corridor to move ahead. Whilst this phenomenon deserves a further in-depth investigation, this result is remarkable as it shows that correlation information provides a fine tool to detect—possibly unforeseen—regularities.

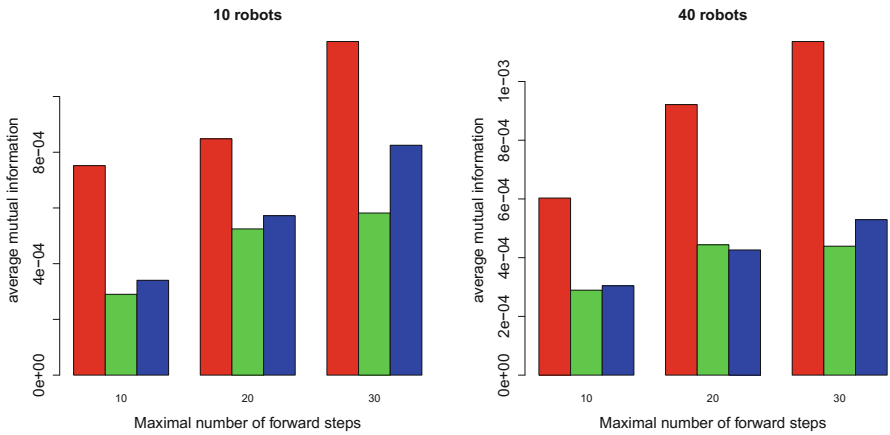


Fig. 5. Mutual information for the dodecagonal arena case, with 1 robot (left) and the 40 robots (right). Maximum turning angles: ■ 40° ■ 90° ■ 180°

A mutual information analysis of robots' trajectories may provide an estimation of the reciprocal influence between robots. Mutual information is computed for all the possible robot pairs and then averaged. The barplots in Fig. 5 show that the interdependence among robots is highest for the case of 40 robots and that the interactions are stronger for smaller turning angles. This analysis is in agreement with the expectations and complements the information gained by the previous metrics.

For completeness, we conclude this section by mentioning the results returned by the application of the set-based complexity. SBC is computed by considering the sequence of swarm states as a set of strings; therefore, it is a measure of the ensemble of robots, rather than of the robots taken individually. Barplots of

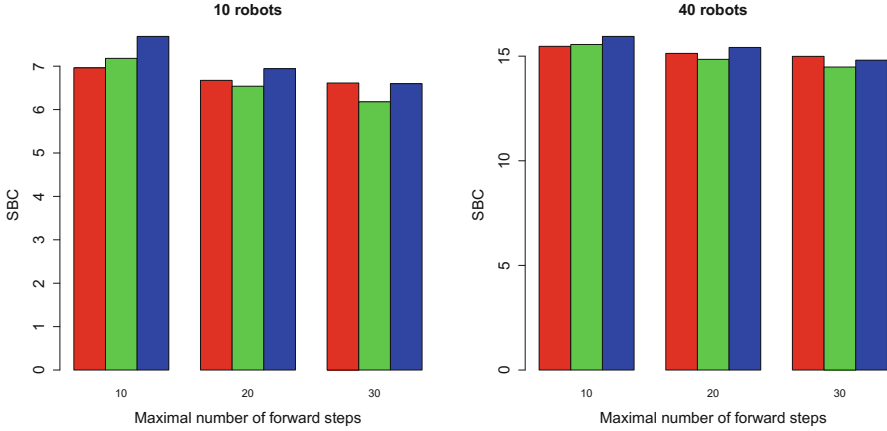


Fig. 6. Set-based complexity for the dodecagonal arena case, with 1 robot (left) and the 40 robots (right). Maximum turning angles: ■ 40° ■ 90° ■ 180°

this analysis are depicted in Fig. 6. We observe that SBC does not differentiate considerably as a function of forward steps and maximum turning angle. Conversely, it is worth to emphasise that the SBC values double moving from 10 to 40 robots, and also that the impact of forward steps number is stronger in the case of 10 robots, where the interference among robots is limited. Nevertheless, as the robots are mainly characterised by random walk, the potential of SBC can not be expressed completely and we expect that this metric could be particularly useful in non-stationary cases.

5 Conclusion and Future Work

The results of this exploratory study show that complexity metrics can capture relevant features, such as patterns, in traces of robot swarm dynamics. We have chosen the most known complexity measures, mainly from information theory, and applied them to a simple task for swarm robotics characterised by a stationary dynamics. As expected, metrics devised for measuring specific dynamical traits return similar results and an heterogeneous selection of them is likely to be the best choice to produce a *complexity fingerprint* of the system. A minimal fingerprint for a stationary case should be composed of metrics focusing on (a) randomness (e.g. LZ complexity), (b) patterns (e.g. block entropy and its derivatives) and (c) interdependence among robots in the swarm (e.g. mutual information).

We are currently enlarging the set of metrics, by including also statistical complexity measures based on model construction, and we plan to apply also local measures [19] and information theoretical measures specifically designed for capturing dynamical properties of the system [31]. Experiments on further stationary cases are planned, such as flocking and memoryless foraging with

random walk. The next step will be to address also non-stationary cases, like e.g. aggregation, so as to be able to tackle swarm missions in which robots may be characterised by changes in their dynamical behaviour. As stated in the introduction, our aim is to devise tools for helping the automatic design of controllers for robot swarms, so our research agenda include as a further step the use of complexity measures both as analysis tool and task-agnostic merit factors.

Acknowledgements. Andrea Roli acknowledges the support of Université libre de Bruxelles as visiting professor in the “Chaire internationale” programme. Mauro Birattari acknowledges support from the Belgian Fonds de la Recherche Scientifique – FNRS. The project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. 681872).

References

1. Ay, N., Bertschinger, N., Der, R., Güttler, F., Olbrich, E.: Predictive information and explorative behavior of autonomous robots. *Eur. Phys. J. B - Condens. Matter Complex Syst.* **63**(3), 329–339 (2008)
2. Badii, R., Politi, A.: *Complexity: Hierarchical Structures and Scaling in Physics*, vol. 6. Cambridge University Press, Cambridge (1999)
3. Birattari, M., Delhaisse, B., Francesca, G., Kerdoncuff, Y.: Observing the effects of overdesign in the automatic design of control software for robot swarms. In: Dorigo, M., Birattari, M., Li, X., López-Ibáñez, M., Ohkura, K., Pinciroli, C., Stützle, T. (eds.) ANTS 2016. LNCS, vol. 9882, pp. 149–160. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44427-7_13
4. Brambilla, M., Ferrante, E., Birattari, M., Dorigo, M.: Swarm robotics: a review from the swarm engineering perspective. *Swarm Intell.* **7**(1), 1–41 (2013)
5. <http://www.bzip.org>. Accessed 30 Nov 2016
6. Cover, T., Thomas, J.: *Elements of Information Theory*. Wiley, Hoboken (2012)
7. Crutchfield, J.: The calculi of emergence: computation, dynamics, and induction. *Physica D* **75**, 11–54 (1994)
8. Edlund, J., Chaumont, N., Hintze, A., Koch, C., Tononi, G., Adami, C.: Integrated information increases with fitness in the evolution of animats. *PLoS Comput. Biol.* **7**(10), e1002236 (2011)
9. Francesca, G., Birattari, M.: Automatic design of robot swarms: achievements and challenges. *Front. Rob. AI* **3**, 29 (2016)
10. Francesca, G., Brambilla, M., Brutschy, A., Trianni, V.: AutoMoDe: a novel approach to the automatic design of control software for robot swarms. *Swarm Intell.* **8**(2), 89–112 (2014)
11. Galas, D., Nykter, M., Carter, G., Price, N.: Biological information as set-based complexity. *IEEE Trans. Inf. Theory* **56**, 667–677 (2010)
12. Gell-Mann, M., Lloyd, S.: Information measures, effective complexity, and total information. *Complexity* **2**(1), 44–52 (1996)
13. Grassberger, P.: How to measure self-generated complexity. *Phys. A: Stat. Mech. Appl.* **140**(1–2), 319–325 (1986)
14. Kolmogorov, A.: Three approaches to the quantitative definition of information. *Prob. Inf. Transm.* **1**(1), 1–7 (1965)

15. Lempel, A., Ziv, J.: On the complexity of finite sequences. *IEEE Trans. Inf. Theory* **22**(1), 75–81 (1976)
16. Li, W.: On the relationship between complexity and entropy for Markov chains and regular languages. *Complex Syst.* **5**(4), 381–399 (1991)
17. Lindgren, K.: *Information theory for complex systems - an information perspective on complexity in dynamical systems, physics, and chemistry*. Chalmers (2014). <http://studycas.com/c/courses/it>
18. Lindgren, K., Nordahl, M.: Complexity measures and cellular automata. *Complex Syst.* **2**(4), 409–440 (1988)
19. Lizier, J.: *The Local Information Dynamics of Distributed Computation in Complex Systems*. Springer Theses Series. Springer, Heidelberg (2013). <https://doi.org/10.1007/978-3-642-32952-4>
20. Lloyd, S.: Measures of complexity: a nonexhaustive list. *IEEE Control Syst. Mag.* **21**(4), 7–8 (2001)
21. Lopez-Ruiz, R., Mancini, H., Calbet, X.: A statistical measure of complexity. *Phys. Lett. A* **209**, 321–326 (1995)
22. Nicolis, G., Nicolis, C.: *Foundations of Complex Systems: Emergence, Information and Prediction*. World Scientific, Singapore (2012)
23. Pinciroli, C., Trianni, V., O’Grady, R., Pini, G., Brutschy, A., Brambilla, M., Mathews, N., Ferrante, E., Di Caro, G., Ducatelle, F., Birattari, M., Gambardella, L., Dorigo, M.: ARGoS: a modular, multi-engine simulator for heterogeneous swarm robotics. *Swarm Intell.* **6**(4), 271–295 (2012)
24. Prokopenko, M., Boschetti, F., Ryan, A.: An information-theoretic primer on complexity, self-organization, and emergence. *Complexity* **15**(1), 11–28 (2009)
25. Prokopenko, M.: *Guided Self-Organization: Inception*, vol. 9. Springer Science & Business Media, Heidelberg (2013). <https://doi.org/10.1007/978-3-642-53734-9>
26. Shalizi, C., Crutchfield, J.: Computational mechanics: pattern and prediction, structure and simplicity. *J. Stat. Phys.* **104**(3), 817–879 (2001)
27. Shannon, C.: A mathematical theory of communication. *Bell Syst. Tech. J.* **27**(1, 2), 379–423, 623–656 (1948)
28. Sperati, V., Trianni, V., Nolfi, S.: Evolving coordinated group behaviours through maximisation of mean mutual information. *Swarm Intell.* **2**(2), 73–95 (2008)
29. Trianni, V.: *Evolutionary Swarm Robotics: Evolving Self-Organising Behaviours in Groups of Autonomous Robots*, vol. 108. Springer, Heidelberg (2008). <https://doi.org/10.1007/978-3-540-77612-3>
30. Utro, F., Di Benedetto, V., Corona, D., Giancarlo, R.: The intrinsic combinatorial organization and information theoretic content of a sequence are correlated to the DNA encoded nucleosome organization of eukaryotic genomes. *Bioinformatics* **32**(6), 835–842 (2015)
31. Villani, M., Roli, A., Filisetti, A., Fiorucci, M., Poli, I., Serra, R.: The search for candidate relevant subsets of variables in complex systems. *Artif. Life* **21**(4), 412–431 (2015)



Identification of “Die Hard” Nodes in Complex Networks: A Resilience Approach

Angela Lombardi¹(✉), Sabina Tangaro², Roberto Bellotti^{2,3},
Angelo Cardellicchio¹, and Cataldo Guaragnella¹

¹ Dipartimento di Ingegneria Elettrica e dell’Informazione,
Politecnico di Bari, Bari, Italy
angela.lombardi@poliba.it

² Istituto Nazionale di Fisica Nucleare, Sezione di Bari, Bari, Italy

³ Dipartimento Interateneo di Fisica “M. Merlin”,
Università degli Studi di Bari “A. Moro”, Bari, Italy

Abstract. The topology of a network defines the structure on which physical processes dynamically evolve. Even though the topological analysis of these networks has revealed important properties about their organization, the components of real complex networks can exhibit other significant characteristics. In this work we focus in particular on the distribution of the weights associated to the links. Here, a novel metric is proposed to quantify the importance of both nodes and links in weighted scale-free networks in relation to their resilience. The resilience index takes into account the complete connectivity patterns of each node with all the other nodes in the network and is not correlated with other centrality metrics in heterogeneous weight distributions.

Keywords: Complex networks · Resilience · Percolation · Centrality
Scale-free networks · Weighted centrality metrics

1 Introduction

Complex networks have become a powerful tool for analyzing interactions in a great variety of contexts [1, 2]. By using the complex networks framework, a system can be modeled in terms of nodes connected by binary or weighted edges whose magnitude quantify respectively the presence or the strength of the links between them. Several graph metrics that are able to characterize the statistical properties of weighted networks combining both topology and weight distributions have been proposed [3]. Weighted networks are particularly interesting for assessing topological properties in systems whereby is critical the importance of connections, e.g., social networks, ecological and biological systems, well-known for their hierarchical organization. As an example, there is clear evidence that many real-world networks exhibit a power law degree

distribution which confers properties of structural robustness amongst attacks or failures [4].

Such properties are strictly related to the concept of percolation, i.e. the existence of a critical probability at which a single connected giant component exists and below which the network is composed of isolated clusters [1]. Different approaches have been proposed to assess the degree of the fragmentation of a network when a finite number of links are removed. Typically, metrics that quantify the importance or centrality of nodes are evaluated as edges are gradually removed from the network [5–9]. Several studies have tested the vulnerability of some synthetic networks for both binary and weighted cases, finding some network topologies more prone to attacks than others.

However, most of the real networks are characterized by a great heterogeneity of topological properties that are only partially taken into account in synthetic simulations. For instance, in weighted networks even weak connections can be statistically significant for a particular structural topology [10]. As an example, synthetic scale-free networks can be generated by using the Barabási-Albert (B-A) algorithm [1, 11]. Accordingly, the resulting networks have a power law degree distribution $P(k) \sim k^{-\gamma}$ with $1 \leq \gamma \leq 3$. However, although the topological properties can highlight many interesting aspects of such real networks, it has been showed that a number of systems with scale-free topology also presents broad distributions of weights and non-trivial correlations between weights and topology structure [3, 12]. Hence, the heterogeneity of the weight distributions should be considered to investigate the complex features of real scale-free networks [13].

Here, a new resilience index is proposed to capture the importance of both nodes and links of a complex network. Specifically, a multidimensional approach is adopted to quantify the importance of nodes and links in relation to their survival rate for progressive removal of links in the network. Weighted undirected networks with scale-free topology are simulated to test the capability of the proposed resilience index to detect the most important nodes and links of the networks. Different weight distributions are considered in order to reflect the heterogeneity of links' strengths in a real scenario. Other centrality metrics known in the literature are compared to show their correlation with the proposed index.

2 Methods

2.1 Synthetic Networks

The B-A algorithm was used to generate synthetic scale-free networks with the same power law degree distribution $P(k) \sim k^{-\gamma}$ with $\gamma = 2$. The mechanisms of growth and preferential attachment are the main features of the scale-free networks: new nodes tend to connect to the more connected nodes that become hubs of the network. Then, in a scale-free topology, most of the nodes are weakly connected and only few strongly connected nodes are the critical hubs of the network.

In particular, in order to represent two different kinds of weight-topology correlations, the weight of the link $w_{i,j}$ between nodes i and j was assigned as:

1. $w_{i,j} = k_i k_j$, to simulate a power law weight distribution setting a full weight-topology correlation.
2. $w_{i,j} \sim \mathcal{U}(0, 1)$, where a random uniform distribution of numbers in the range $[0, 1]$ is introduced to remove the correlation between weights and topology structure.

Figure 1 shows the same topology of a scale-free network composed by $N = 100$ nodes with the aforementioned weight distributions. As it can be seen, the most significant link in the power law case is established between the two hubs of the network (see Fig. 1(a)).

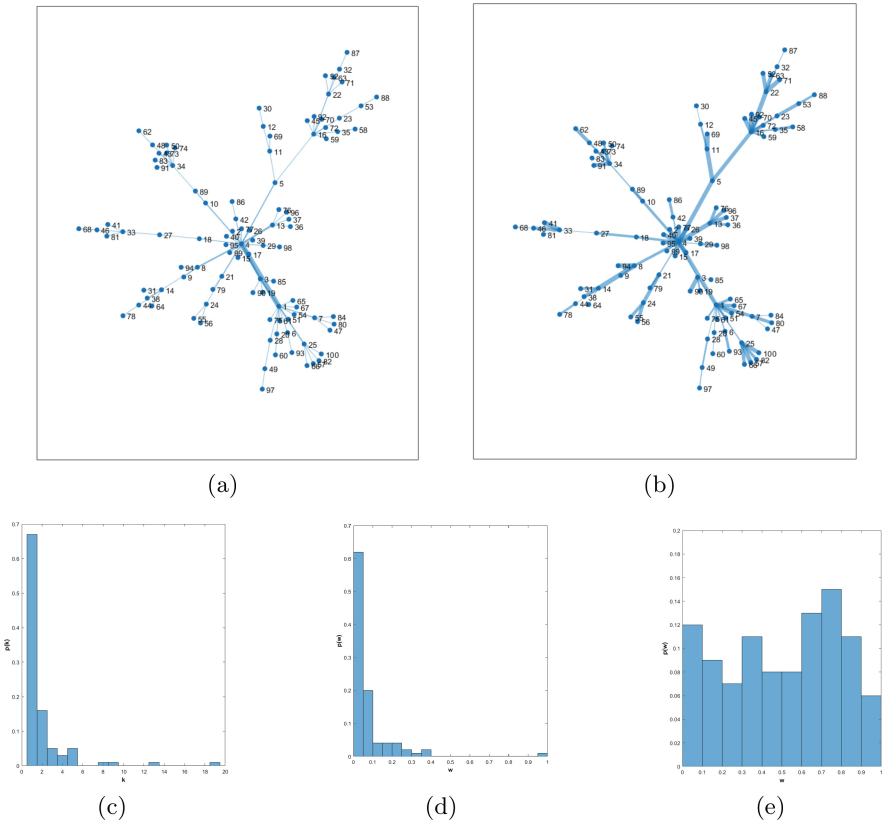


Fig. 1. A scale-free network topology composed by $N = 100$ nodes: (a) power law weight graph; (b) random uniform weight graph; (c) degree distribution; (d) power law weight distribution; (e) random uniform weight distribution. Line width of the graph links are proportional to their weights.

2.2 Centrality Metrics

In an undirected network $G(V, E)$, V and E being respectively the set of nodes and set of links, the importance of a node was assessed by using the following metrics:

- the degree k_i of a node v_i , i.e, simply the number of nodes attached to v_i :

$$k_i = \sum_{j=1}^N a_{ij} \tag{1}$$

where N is the number of the nodes of the network and $A = \{a_{ij}\}$ is the adjacency matrix, with $a_{ij} = 1$ if nodes v_i and v_j are connected and 0 otherwise. Degree represents the simplest centrality index as it assigns more importance to more connected nodes which have more influence over their neighbors.

- The strength of a node v_i , defined as the sum of the weights of the links associated to v_i :

$$s_i = \sum_{j=1}^N w_{ij} \tag{2}$$

where $W = \{w_{ij}\}$ is the weighted adjacency matrix, with $w_{ij} > 0$ if v_i and v_j are connected and $w_{ij} = 0$ if they are not connected.

- The centrality of a node can also be related to its position in the network with respect to the paths of information flow. A path from v_i to v_j is a sequence of vertices and edges, such that each edge connects its preceding with its succeeding vertex. In weighted networks, a cost criterion should be specified in order to associate weights to distances. In a general context, link weights usually do not represent the costs of connections, but their strength, so the reciprocal of weights can be directly related to their distance paths without loss of generality [14]. Let $d_G(i, j)$ be the distance between vertices v_i and v_j , i.e. the length of the shortest path among all the paths connecting v_i and v_j that can be computed for example by using Dijkstra’s algorithm [15]. Closeness centrality quantify the proximity of each node to the rest of the network and it is expressed as:

$$CC_i = \frac{1}{\sum_{j=1}^N d_G(i, j)} \tag{3}$$

A high value of closeness means that a node is easily reached from all the other nodes with few steps.

- Betweenness centrality, expressed as the fraction of the shortest paths that pass through each node or edge [16]:

$$BC_i = \sum_{i \neq j \neq t} \frac{\sigma_{jt}(i)}{\sigma_{jt}} \tag{4}$$

where σ_{jt} denotes the number of shortest paths from v_j to v_t and $\sigma_{jt}(i)$ denotes the number of shortest paths from v_j to v_t that pass through v_i .

Betweenness highlights nodes (or edges) that, upon removal, would affect efficient routing across the network.

For both closeness and betweenness metrics the length of the shortest path $d_G(i, j)$ between vertices v_i and v_j , is defined as:

$$d_G(i, j) = \min \left(\frac{1}{w_{ih}} + \dots + \frac{1}{w_{hj}} \right) \tag{5}$$

where h are intermediary nodes on paths between the two vertices v_i and v_j .

- The eigenvector centrality, i.e., an iterative centrality in which the influence of a node is determined by the number and influence of its neighbors [17]:

$$EIG_i = \frac{1}{\lambda} \sum_{j=1}^N w_{ij} EIG_j \tag{6}$$

Where λ is the largest eigenvalue in absolute value of W .

2.3 Resilience Index

The computation of the resilience index requires the following steps which are also shown in Fig. 2:

1. given the adjacency matrix W of the network, in which the entry (i, j) indicates the weight of the link between the node i and j (w_{ij}), the range of all the weights is divided into L levels. In this step, L percolation levels are identified. Ideally, a dense range of levels should be considered to take into account all possible percolation scenarios;
2. the matrix W is incrementally thresholded by removing all the links of the network whose weight is below the threshold at each of the L levels;
3. a multilayer matrix T is defined where the entry (i, j, l) represents the weight of the link between the nodes i and j at the l^{th} level of percolation;
4. the connectivity pattern of the node i for the l^{th} level of percolation is defined as:

$$P_{i,l} = T_{i,j,l} \quad j = 1, \dots, N. \tag{7}$$

5. the similarity between connectivity patterns of each couple of nodes at each level of percolation is expressed as their cosine similarity:

$$D_{ij,l} = \frac{P_{i,l} \cdot P_{j,l}}{\|P_{i,l}\|_2 \|P_{j,l}\|_2}. \tag{8}$$

6. $D_{ij,l} \neq 0$ for a certain degree of similarity between the connectivity patterns of the nodes i and j and it is set $D_{ij,l} = 0$ if the node i or the node j becomes disconnected. Similarly, $D_{ii,l} = 1$ if the node i is connected at least with another node of the network and $D_{ii,l} = 0$ if it becomes completely isolated from the rest of the network.

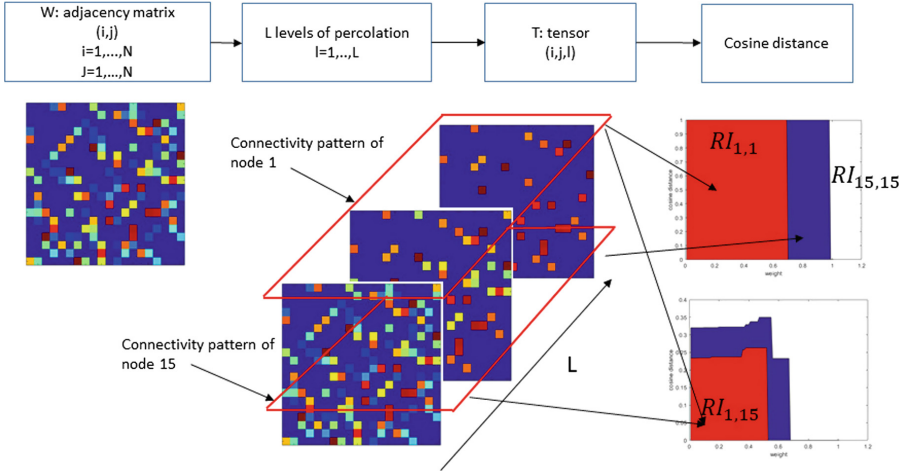


Fig. 2. Steps required to assess the resilience index. The weighted adjacency matrix is percolated into L levels and a tensor T is composed of the matrices resulting from each percolation level. The cosine distance between couple of connectivity patterns of the nodes is computed to assess a percolation curve. Finally the resilience index is expressed as the area under the curve.

The resilience index of the link (RI_{link}) between the nodes i and j is defined as the area under the curve $D_{ij} = D_{ij,l}$, $l = 1, \dots, L$:

$$RI_{ij} = \sum_{l=0}^L D_{ij,l} \tag{9}$$

Likewise, the resilience index of the node (RI_{node}) i is expressed as:

$$RI_i = \sum_{l=0}^L D_{ii,l} \tag{10}$$

2.4 Correlation Analysis

In this work, both a simple example involving the two networks shown in Fig. 1 and a simulation study are presented to investigate the importance of nodes and links of the networks.

For each of the two networks were computed:

- the following node centrality metrics: degree (K); strength (S); betweenness (BC); closeness (CC); eigenvector centrality (EIG);
- the edge betweenness (EB) as link metric.

Then, a correlation analysis was carried out to:

- investigate the presence of possible correlations between the proposed index and the all the other centrality metrics;
- examine the ranking mechanism of each nodal centrality metric with respect to the two weight distributions. To this end, Pearson’s correlation coefficient was computed for the two weight distributions for each nodal metric. The correlation coefficient has a range $-1 \leq r \leq 1$, where the coefficient has value 1 for perfect ranking, value -1 for anti-correlation (i.e., one ranking is the reverse of the other) and value 0 for two uncorrelated rankings.

Numerical simulations of 100 scale-free networks with the same parameters, were conducted to generalize the results. In particular, for each of the 100 simulated network, the weights of the starting power-law distribution were progressively randomized until a completely random final configuration. The correlation analysis between the proposed index and each of the known metrics was carried out for each randomization interval of the weights, while the comparison of the node rankings was only performed between the initial and the final configuration for each nodal metric.

3 Results

The centrality metrics evaluated for each node of the two networks are shown in Fig. 3. Obviously, the degree function is the same for both networks because they have the same structural topology. Strength and eigenvector values are emphasized for both hub nodes of the network (nodes 1 and 4 as it can be noted in Fig. 1) and for few other nodes with degree greater than that of the “leaf” nodes, i.e., those nodes with just one link. The values of betweenness are slight different only for the two hubs, so this metric seems to not take almost into account the distribution of the weights. On the other hand, some nodes with few connections exhibit high values of the resilience index for the random weight distribution, while its trend is closely correlated with that of the eigenvector centrality for the weight power law. The behavior of closeness centrality is clearly different in the two networks, and it is considerably lower for all nodes of the network with random weight distribution. However, there is no apparent relationship between the centrality values and the role of nodes (hubs vs. leaf nodes).

The results of the correlation analysis between the proposed metric and the other centrality measures are listed in Table 1. High correlation values are observed for all centrality metrics except for the closeness centrality in the network with scale-free weight distribution; whereas there are low correlations (<0.4) between the resilience index and degree, betweenness and eigenvector centrality respectively in the random weight distribution network. Strength and the proposed index seem to exhibit the highest value of correlation in the latter network, while closeness is negatively correlated, even if with low correlation index. Moreover, there is no correlation between the resilience index defined for the links and the edge betweenness for both weight distributions.

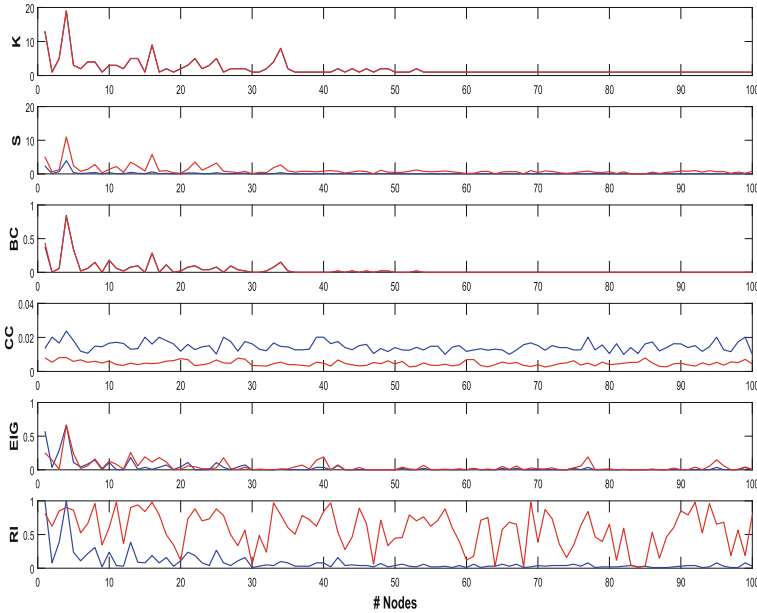


Fig. 3. Values of the centrality metrics (degree, strength, betweenness, closeness, eigenvector and resilience index) for each of the 100 nodes for the network with the power law weight distribution (blue) and that with the random uniform weight distribution (red) reported in Fig. 1. (Color figure online)

Table 1. Correlation between the resilience index and the other centrality metrics ($p < 0.0001$).

Network	K	S	BC	CC	EIG	EB
Power law	0.88	0.93	0.85	0.23	0.98	-0.0069
Uniform	0.32	0.50	0.29	-0.19	0.38	-0.0171

The rank correlation for each centrality metric is shown in Table 2. Strength, betweenness (both nodal and edge) and eigenvector centralities display significant high correlation values between the ranking of the nodes (and the links for EB) in the two weight distributions, whilst closeness and resilience index are significantly dissimilar in ranking nodes. Although with a higher correlation value, the RI_{link} metric confirms the same behavior of the RI_{node} .

Figure 4 shows the evolution of the correlation between the RI_{node} and the other nodal metrics and between RI_{link} and edge betweenness as a function of the percentage of randomization of the weights for the 100 simulated networks starting from a configuration with scale-free weight distribution. All nodal metrics except CC , are highly correlated with the resilience index RI_{node} for low randomized of the weights and then correlation values decrease until they converge around the median value $r_m = 0.3$ for K, BC and EIG and $r_m = 0.5$

Table 2. Correlation coefficient (r) with p-value (p) resulting from correlation between the rankings of the two weight distributions for each centrality metric.

	S	BC	CC	EIG	EB	RI_{node}	RI_{link}
r	0.87	0.99	0.18	0.98	0.98	0.28	0.45
p	<0.0001	<0.0001	0.007	0.38	<0.0001	0.003	<0.0001

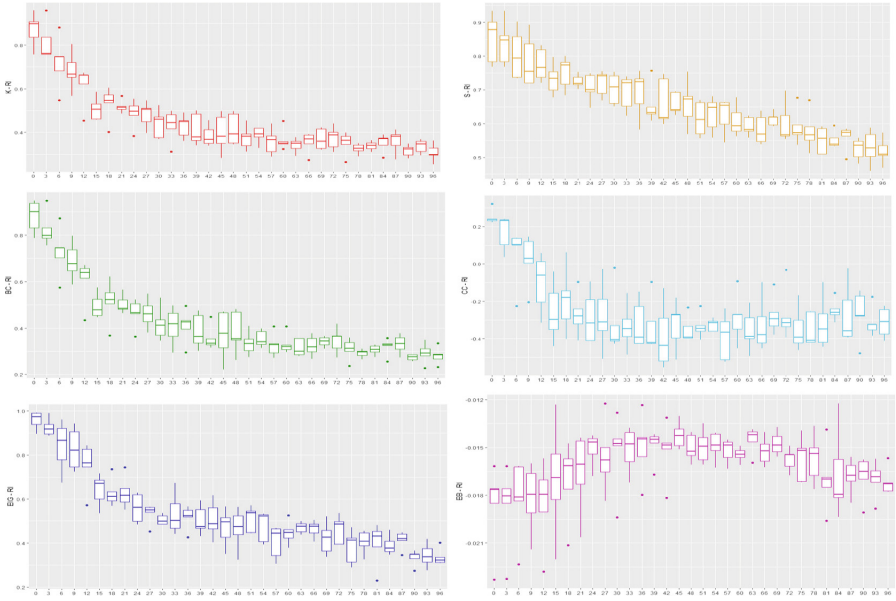


Fig. 4. Boxplots of correlation between the proposed index and each of the nodal centrality metrics for progressive randomization of the weights for the 100 simulated scale-free networks.

for S . The closeness centrality exhibits a median correlation value $r_m = 0.24$ at the scale-free weight distribution, decreasing to negative correlation values with median $r_m = -0.3$. RI_{link} and EB are almost uncorrelated for all percentages of randomization.

The rank correlation analysis on simulated networks highlights that both resilience metrics score the nodes and links of the two extreme network configurations with different importance scales. Indeed, as shown in Fig. 5, they have the lowest correlation values; in contrast, both the betweenness metrics are insensitive to the distribution of weights, showing very high correlation between the ranking mechanisms in the two cases. S and EIG also have significantly higher correlation values (respectively median $r_m = 0.9$ and $r_m = 0.8$), while CC exhibits lower values (median $r_m = 0.48$), confirming a different ranking of the nodes belonging to the two weight distributions.

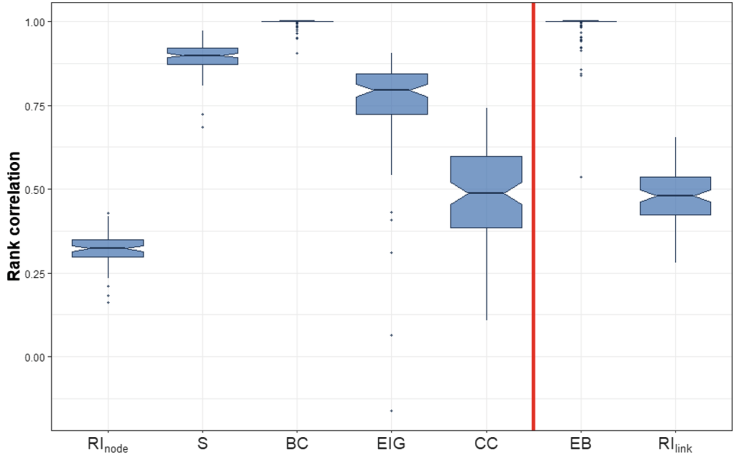


Fig. 5. Boxplots of correlation between node rankings (for the metrics: RI_{node} , S , BC , EIG and CC) and link rankings (for the metrics: EB and RI_{link}) of the initial configuration (i.e., with power-law weight distribution) and the final configuration (i.e., random uniform weight distribution) for the 100 simulated scale-free networks.

The results of the correlation analysis between the proposed metric and the other centrality measures are listed in Table 1.

4 Discussion

The statistical analysis carried out highlights an interesting phenomenon: when the topology and the weight distribution are correlated, the proposed metric is not different from the others and resilient nodes are also central (and vice versa). However, when this correlation is removed, the proposed index is able to provide more information about the position of a node in relation to the network. The resilience index considers complete connectivity patterns of each node with the rest of the network at varying degrees of percolation. For this reason, even leaf nodes strongly connected to a particularly resilient node can also be resilient nodes. This aspect is clearly visible in the scale free networks with random weight distribution where weights are not assigned according to the underlying topological structure and even a peripheral node may have a strong connection with a hub node.

Real scale free networks with variable weight distributions have been identified and examined [3, 12, 13, 18]. In particular, criteria for both model and classify networks in which the connectivity of the node does not affect the weights of the links and networks in which the connectivity strongly influence them, have been reported [19]. Several measures to characterize weighted networks have been proposed, but they have not been tested in this context so far. It is certainly true that identifying important nodes is not trivial. First of all, because there is no a

universal method for quantifying the importance of a node. The definition of centrality varies according to the context in which a specific metric is applied. In [20] some factors to evaluate each centrality metric are suggested: radial (e.g., degree, closeness, and eigenvector centrality) and medial measures (e.g., betweenness centrality) are defined according to the information flow through the network and number of walks. A centrality metric should identify the role of the node in relation to the global characteristics of the network and not simply on the basis of the topology. The results of the rank correlation analysis show that some metrics assign very similar scores in the two situations. It is worth nothing the case of the betweenness centrality according to which the two networks seem to be completely similar. These findings could be due to the fact that some indices take into account only local information of a node. The proposed metric differs from all other centrality definitions since multidimensional patterns of connectivity are considered for its computation. It is also important to note that the resilience index should not be considered more effective than the other metrics just because it is able to better discriminate the two types of considered weight distribution, but that integrating the information provided by the other metrics with that of the proposed index could lead to new centrality metrics and reach a higher accuracy.

5 Conclusion

In this work, a novel metric is proposed to evaluate the importance of nodes and links in weighted networks in relation to their resilience. The proposed metric is applied to weighted scale-free complex networks with different weight-topology correlations. Other centrality metrics that assess statistical properties of weighted networks combining both topology and weight distributions have been used and compared to the proposed resilience index. Although these measures allow to quantify the centrality, cohesiveness and influence of a node in a complete and heterogeneous way, they do not consider the dynamic evolution of the network. The proposed index takes into account the complete connectivity patterns of each node with all the other nodes in the network and the correlation analysis shows that it is not related with other centrality metrics when the correlation between the topology and the weight distribution of the scale-free networks is removed. In future work, an accurate analysis of the influence of the choice of different percolation intervals on the final performances will be carried out; additionally, more effective metrics of centrality could be defined by integrating such new index with the known centrality measures on real instances of networks.

References

1. Albert, R., Barabási, A.L.: Statistical mechanics of complex networks. *Rev. Mod. Phys.* **74**(1), 47 (2002)
2. Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.: Complex networks: structure and dynamics. *Phys. Rep.* **424**(4), 175–308 (2006)

3. Barrat, A., Barthelemy, M., Pastor-Satorras, R., Vespignani, A.: The architecture of complex weighted networks. *Proc. Natl. Acad. Sci. U.S.A.* **101**(11), 3747–3752 (2004)
4. Barabási, B.A.L., Bonabeau, E.: Scale-free. *Sci. Am.* **288**(5), 50–59 (2003)
5. Holme, P., Kim, B.J., Yoon, C.N., Han, S.K.: Attack vulnerability of complex networks. *Phys. Rev. E* **65**(5), 056109 (2002)
6. Gol'dshtein, V., Koganov, G., Surdutovich, G.I.: Vulnerability and hierarchy of complex networks. arXiv preprint [arXiv:cond-mat/0409298](https://arxiv.org/abs/cond-mat/0409298) (2004)
7. Latora, V., Marchiori, M.: Vulnerability and protection of infrastructure networks. *Phys. Rev. E* **71**(1), 015103 (2005)
8. Boccaletti, S., Buldú, J., Criado, R., Flores, J., Latora, V., Pello, J., Romance, M.: Multiscale vulnerability of complex networks. *Chaos: Interdisc. J. Nonlinear Sci.* **17**(4), 043110 (2007)
9. Mishkovski, I., Biey, M., Kocarev, L.: Vulnerability of complex networks. *Commun. Nonlinear Sci. Numer. Simul.* **16**(1), 341–349 (2011)
10. Granovetter, M.S.: The strength of weak ties. *Am. J. Sociol.* **78**(6), 1360–1380 (1973)
11. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. *Science* **286**(5439), 509–512 (1999)
12. Barthélemy, M., Barrat, A., Pastor-Satorras, R., Vespignani, A.: Characterization and modeling of weighted networks. *Phys. A: Stat. Mech. Appl.* **346**(1), 34–43 (2005)
13. Newman, M.E.: Analysis of weighted networks. *Phys. Rev. E* **70**(5), 056131 (2004)
14. Opsahl, T., Agneessens, F., Skvoretz, J.: Node centrality in weighted networks: generalizing degree and shortest paths. *Soc. Netw.* **32**(3), 245–251 (2010)
15. Brandes, U.: A faster algorithm for betweenness centrality. *J. Math. sociol.* **25**(2), 163–177 (2001)
16. Freeman, L.C.: A set of measures of centrality based on betweenness. *Sociometry* **40**(1), 35–41 (1977)
17. Bonacich, P.: Power and centrality: a family of measures. *Am. J. Sociol.* **92**(5), 1170–1182 (1987)
18. Dall'Asta, L., Barrat, A., Barthélemy, M., Vespignani, A.: Vulnerability of weighted networks. *J. Stat. Mech: Theory Exp.* **2006**(04), P04006 (2006)
19. Bianconi, G.: Emergence of weight-topology correlations in complex scale-free networks. *EPL (Europhys. Lett.)* **71**(6), 1029 (2005)
20. Borgatti, S.P., Everett, M.G.: A graph-theoretic perspective on centrality. *Soc. Netw.* **28**(4), 466–484 (2006)

Optimization



Automatic Algebraic Evolutionary Algorithms

Marco Baioletti¹, Alfredo Milani^{1,2}, and Valentino Santucci¹(✉)

¹ Department of Mathematics and Computer Science,
University of Perugia, Perugia, Italy

{marco.baioletti,alfredo.milani,valentino.santucci}@unipg.it

² Department of Computer Science, Hong Kong Baptist University,
Kowloon Tong, Hong Kong

Abstract. Motivated from the previously proposed algebraic framework for combinatorial optimization, here we introduce a novel formal languages-based perspective on discrete search spaces that allows to automatically derive algebraic evolutionary algorithms. The practical effect of the proposed approach is that the algorithm designer does not need to choose a solutions encoding and implement algorithmic procedures. Indeed, he/she only has to provide the group presentation of the discrete solutions of the problem at hand. Then, the proposed mechanism allows to automatically derive concrete implementations of a chosen evolutionary algorithms. Theoretical guarantees about the feasibility of the proposed approach are provided.

Keywords: Algebraic evolutionary algorithms
Combinatorial optimization · Formal language perspective

1 Introduction

In a previous series of articles [1, 3, 11, 13], we have introduced an abstract algebraic framework for combinatorial optimization problems. The framework allows to encode in algebraic terms the geometry of the search moves performed by a large class of evolutionary algorithms on the search space of combinatorial problems.

Concrete implementations of the framework have been proposed for discrete spaces such as the permutations and bit-string spaces. Hence, algebraic evolutionary algorithms, such as algebraic differential evolution and particle swarm optimization, have been proposed [1, 11]. Interestingly, state-of-the-art and very competitive results have been obtained for permutation flowshop scheduling problems [11, 12] and linear ordering problems [2, 3].

The main achievement of the algebraic framework is the proposal of abstract definitions for operators that allow to combine and operate on the discrete solutions of the problem at hand. In particular, the proposed operations are addition, subtraction and scalar multiplication. Some abstract algebraic and

geometric properties, derived from group theory, guarantee that their effects on the involved discrete solutions are geometrically similar to what happen in the classical Euclidean space.

However, the definitions are merely abstract and the algorithm designer needs to instantiate them for any finitely generated group at hand. For instance, randomized decomposer have to be (and have been) provided for the groups of permutations and bit-strings. As an additional, though secondary, result, here we also show how the search space of integer vectors can be represented in the framework.

In this paper we further evolve the framework by proposing general implementations of the abstract operators that are no more abstract but directly operative on any search space that respect some conditions, i.e., to be representable by a finitely presented group. To achieve this aim, we consider a formal language-perspective directly derived from advanced group theory concepts. Therefore, we provide a mechanism to automatically derive operative and universal implementations of the previously proposed algebraic operators by exploiting the concept of group presentation. Discrete solutions are represented as strings of an alphabet (of generators). By changing the alphabet and the equivalence relations on the strings, it is possible to use the Knuth-Bendix completion algorithm [9] to automatically derive concrete operators on different types of solutions (permutations, bit-strings, etc.).

Practically, we make easy the work of the algorithm designer that can now avoid to choose a solutions encoding and implement the abstract procedures of the framework for this encoding. Note anyway, that this proposal is a sort of “proof of concept”. Indeed, we do not provide any experimental result, but only theoretical guarantees about the feasibility of the proposed implementations.

The rest of the paper is organized as follows. Section 2 describes the previously proposed abstract algebraic framework together with some of its concrete implementations. The algebraic evolutionary operators are then derived in Sect. 3. The core of the paper is represented by Sects. 4 and 5 where we provide, respectively, theoretical foundations of the language-based perspective, and the concrete and general algorithmic implementations. Finally, Sect. 6 concludes the paper by also providing some future lines of research.

2 Abstract Algebraic Framework

In this section we provide a concise description of the algebraic framework for evolutionary computation previously proposed in [11], together with its extension introduced in [3]. The framework is based on the notion of *finitely generated group* and the related algebraic and geometric concepts. Its aim is to introduce the operations \oplus , \ominus , \odot on the set of discrete solutions in such a way that they simulate, as much as possible, the analogous vector operations of the Euclidean space.

2.1 Search Spaces and Finitely Generated Groups

The triplet $G = (X, \star, H)$ is a finitely generated group representing a combinatorial search space if and only if:

- X is the discrete set of solutions in the search space;
- $\star : X \times X \rightarrow X$ is a binary operation on X which satisfies the group properties: associativity, existence of the identity $e \in X$, and existence of the inverse $x^{-1} \in X$ for any $x \in X$; if \star is also commutative, the group is Abelian, but it is not required;
- $H \subseteq X$ is a finite generating set of the group, i.e., any $x \in X$ can be decomposed as $x = h_1 \star \dots \star h_l$ for some $h_1, \dots, h_l \in H$.

A decomposition $x = h_1 \star \dots \star h_l$ of $x \in X$ is minimal if there exists no other decomposition $x = h'_1 \star \dots \star h'_m$ with $m < l$. The length l of a minimal decomposition of x is the weight of x and it is denoted by $|x|$.

Given a finitely generated group $G = (X, \star, H)$, its Cayley graph $\mathcal{C}(G)$ is the labelled digraph whose vertexes are the solutions in X and there exists an arc from x to y labelled by $h \in H$ if and only if $y = x \star h$.

In the Cayley graph, for all $x \in X$, every directed path from e to x corresponds to a decomposition of x : if the arcs labels occurring in the path are $\langle h_1, h_2, \dots, h_l \rangle$, then $x = h_1 \star h_2 \star \dots \star h_l$. As a consequence, shortest paths from e to x correspond to minimal decompositions of x . More generally, a shortest path from x to y , where $x, y \in X$, corresponds to a minimal sequence of generators $\langle h_1, h_2, \dots, h_l \rangle$ such that $x \star (h_1 \star h_2 \star \dots \star h_l) = y$. Hence, $\langle h_1, h_2, \dots, h_l \rangle$ is a minimal decomposition of $x^{-1} \star y$.

The diameter D of $\mathcal{C}(G)$ is defined as the maximal weight of the elements in X . Moreover, an interesting partial order relation, which will be useful later, is defined as follows. For $x, y \in X$, $x \sqsubseteq y$ if and only if there exists (at least) a shortest path from e to y passing by x . For the sake of presentation, here we focus on groups with a unique maximal weight element ω such that $x \sqsubseteq \omega$ for all $x \in X$. The concrete group considered later belongs to such a class.

The Cayley graph has an important geometric interpretation. Indeed, a sequence of generators $\langle h_1, h_2, \dots, h_l \rangle$ can be seen as a *vector* which connects a starting *point* $x \in X$ to the end *point* $y = x \star (h_1 \star h_2 \star \dots \star h_l)$. On the other hand, any element $x \in X$ can be decomposed as a sequence of generators $\langle h_1, h_2, \dots, h_l \rangle$ and therefore it can be considered also as a *free vector*. The dichotomous interpretation of the elements of X , as points and as vectors, allows to define the operations \oplus, \ominus, \odot on X which simulate the analogous operations of the Euclidean space.

2.2 Addition and Subtraction

The addition $z = x \oplus y$ is defined as the application of the vector $y \in X$ to the point $x \in X$. The result z is computed by choosing a decomposition $\langle h_1, h_2, \dots, h_l \rangle$ of y and by finding the end point of the path which starts from x and whose arcs labels are $\langle h_1, h_2, \dots, h_l \rangle$, i.e., $z = x \star (h_1 \star h_2 \star \dots \star h_l)$. By noting

that $h_1 \star h_2 \star \dots \star h_l = y$, the addition \oplus is independent from the generating set and is uniquely defined as

$$x \oplus y := x \star y. \tag{1}$$

Continuing the analogy with the Euclidean space, the difference between two points is a vector. Given $x, y \in X$, the difference $y \ominus x$ produces the sequence of labels $\langle h_1, h_2, \dots, h_l \rangle$ in a path from x to y . Since $h_1 \star h_2 \star \dots \star h_l = x^{-1} \star y$, we can replace the sequence of labels with its product, thus making the difference independent from the generating set. Therefore, \ominus is uniquely defined as

$$y \ominus x := x^{-1} \star y. \tag{2}$$

Both \oplus and \ominus , like their numerical counterparts, are consistent to each other. Indeed, $x \oplus (y \ominus x) = y$ for all $x, y \in X$. Moreover, both operations are not commutative (unless the group is Abelian), \oplus is associative, and e is its neutral element.

2.3 Scalar Multiplication

Again, as in the Euclidean space, it is possible to multiply a vector by a non-negative scalar. Given $a \geq 0$ and $x \in X$, we denote their multiplication with $a \odot x$.

We first provide the conditions that $a \odot x$ has to verify in order to simulate, as much as possible, the scalar multiplication of vector spaces:

- (C1) $|a \odot x| = \lceil a \cdot |x| \rceil$;
- (C2) if $a \in [0, 1]$, $a \odot x \sqsubseteq x$;
- (C3) if $a \geq 1$, $x \sqsubseteq a \odot x$.

Clearly, the scalar multiplication of \mathbb{R}^n satisfies the slight variant of (C1) where the Euclidean norm replaces the group weight and the ceiling is omitted. Besides, similarly to scaled vectors in \mathbb{R}^n , (C2) and (C3) intuitively encode the idea that $a \odot x$ is the element x scaled down or up, respectively.

It is important to note that, fixed a and x , there may be more than one element of X satisfying (C1–C3). This is a clear consequence of the non uniqueness of the minimal decomposition of x . Therefore, different strategies can be devised to compute $a \odot x$. Nevertheless, our aim is to apply the operation in evolutionary algorithms, therefore we denote with $a \odot x$ a randomly selected element satisfying (C1–C3).

Note also that the diameter D induces an upper bound on the possible values for the scalar a . Indeed, for any $x \in X$, let $\bar{a}_x = \frac{D}{|x|}$, if $a > \bar{a}_x$, (C1) would imply $|a \odot x| > D$, but this is impossible. Therefore, similarly to out-of-bounds handling techniques of continuous evolutionary algorithms, we define

$$a \odot x := \bar{a}_x \odot x, \text{ when } a > \bar{a}_x. \tag{3}$$

The multiplication $a \odot x$ can be computed by: (i) randomly selecting a shortest path from e to ω passing by x , and (ii) composing the first $\lceil a \cdot |x| \rceil$ generators

on its arcs. Since any sub-path of a shortest path is itself a shortest path, and by also considering that shortest paths correspond to minimal decompositions, it is easy to see that the conditions (C1–C3) are satisfied.

Let $l = |x|$, we can observe that the sequence of generators $\langle h_1, \dots, h_l, \dots, h_D \rangle$ on the chosen shortest path can be divided in two parts: $\langle h_1, \dots, h_l \rangle$ and $\langle h_{l+1}, \dots, h_D \rangle$. The former is a minimal decomposition of x , while the latter minimally decomposes $x^{-1} \star \omega$. Operatively, only one of the sub-paths is used to compute $a \odot x$. When $a \leq 1$, the generators to compose are all in the first sub-path $\langle h_1, \dots, h_l \rangle$. Conversely, for $a > 1$, it is sufficient to take the first $\lceil a \cdot l \rceil - l$ generators in the second sub-path $\langle h_{l+1}, \dots, h_D \rangle$ and compose them to the right of x .

The pseudo-codes of the two procedures for $a \in [0, 1]$ and $a > 1$ are reported, respectively, in Figs. 1 and 2. Both rely on the abstract procedure *RandDec* which is assumed to return a random minimal decomposition of the element in input. An implementation of *RandDec* has to consider the particularities of the concrete finitely generated group at hand. Note also that *Extend* implements Eq. (3).

```

1: function TRUNCATE( $a \in [0, 1], x \in X$ )
2:    $s \leftarrow \text{RandDec}(x)$ 
3:    $l \leftarrow \text{Length}(s)$ 
4:    $k \leftarrow \lceil a \cdot l \rceil$ 
5:    $z \leftarrow e$ 
6:   for  $i \leftarrow 1$  to  $k$  do
7:      $z \leftarrow z \star s_i$ 
8:   end for
9:   return  $z$ 
10: end function

```

Fig. 1. Truncation algorithm for computing $a \odot x$ when $a \in [0, 1]$

```

1: function EXTEND( $a > 1, x \in X$ )
2:    $s \leftarrow \text{RandDec}(x^{-1} \star \omega)$ 
3:    $l \leftarrow D - \text{Length}(s)$ 
4:    $\bar{a}_x = \frac{D}{l}$ 
5:    $a \leftarrow \min\{a, \bar{a}_x\}$ 
6:    $k \leftarrow \lceil a \cdot l \rceil$ 
7:    $z \leftarrow x$ 
8:   for  $i \leftarrow 1$  to  $k - l$  do
9:      $z \leftarrow z \star s_i$ 
10:  end for
11:  return  $z$ 
12: end function

```

Fig. 2. Extension algorithm for computing $a \odot \pi$ when $a > 1$

2.4 Concrete Implementations

Given a concrete finitely generated group (FGG) modeling the search space at hand, in order to implement the abstract vector operations described in Sects. 2.2 and 2.3, it is sufficient to provide procedures to: (i) invert an element (x^{-1}), (ii) compose two elements ($x \star y$), (iii) randomly decompose an element in terms of the generators (*RandDec*). Moreover, note that the procedures for (i) and (ii) are usually straightforward.

Three concrete FGGs that allow to cover the vast majority of the combinatorial optimization problems are: the group of the n -length bit-strings \mathbb{B}^n , the group of the n -length permutations \mathcal{S}_n , and the group of the n -length integer vectors \mathbb{Z}^n .

The bit-strings in \mathbb{B}^n form a group by considering the classical bitwise XOR operator \vee . The generators are the strings with one 1-bit and $n-1$ 0-bits. Therefore, computing a random decomposition of a given bit-string simply reduces to choosing an ordering (i.e., a permutation) of its 1-bits. Note also that, given a generic $x \in \mathbb{B}^n$ and the generator u_i (i.e., the bit-string with only one 1-bit at position i), the composition $x \vee u_i$ practically corresponds to flip the i -bit of x . Hence, the induced Cayley graph and distance correspond to classical concepts such as, respectively, the binary hypercube and the Hamming distance.

All the permutations of the set $[n] = \{1, \dots, n\}$ form the “symmetric group” \mathcal{S}_n by considering the classical permutation composition operator \circ defined as $(\pi \circ \rho)(i) = \pi(\rho(i))$ for all items $i \in [n]$ and $\pi, \rho \in \mathcal{S}_n$. Different generating sets are possible in \mathcal{S}_n (see [3, 14]). The simplest is the subset of the $n-1$ simple transpositions, i.e., the set $ST = \{\sigma_i \in \mathcal{S}_n : 1 \leq i < n\}$ where σ_i is defined as: $\sigma_i(i) = i+1$, $\sigma_i(i+1) = i$, and $\sigma_i(j) = j$ for $j \in [n] \setminus \{i, i+1\}$. Given a generic $\pi \in \mathcal{S}_n$, the composition $\pi \circ \sigma_i$ corresponds to swap the adjacent items i and $i+1$ in π . Hence, by modifying the classical bubble sort algorithm, in [11] we have provided a randomized decomposer for \mathcal{S}_n . Moreover, other interesting generators are those which encode exchange and insertion moves of generic items in the permutation. Implementations of these generating sets have been discussed and provided in [3].

Finally, the integer vectors in \mathbb{Z}^n form a group by considering the classical arithmetic addition $+$. In this case, the generators are the n -length vectors formed by $n-1$ zeros and one component equal to ± 1 . A randomized decomposer for \mathbb{Z}^n is straightforward to derive. Note also that this group, differently from the other ones, is infinite and it does not have a maximum weight element. Apparently, this does not allow to implement the algorithm *Extend* of Fig. 2. Anyway, a simple generalization of *Extend* fixes the problem. The idea is to iteratively choose a random generators among all the generators that increase the group weight of the current element. In \mathbb{Z}^n , the group weight is the arithmetic sum of the vector components.

3 Algebraic Evolutionary Operators

It is possible to straightforwardly derive algebraic evolutionary operators by using the operations introduced in Sect. 2 in order to redefine the move equations of the most popular evolutionary and swarm intelligence algorithms for continuous optimization.

Here we provide the formal redefinitions for: the mutation operator of Differential Evolution (DE) [15], the velocity and position update equations of Particle Swarm Optimization (PSO) [8], and the update equation of the Firefly Algorithm (FA) [18]. The first two have been proposed in, respectively, [1, 11], while the third is a novelty of this work. The following definitions subsume that a finitely generated group X is given.

The differential mutation of DE, given three distinct population individuals $x_0, x_1, x_2 \in X$ and a scalar $F \in [0, 1]$, generates a mutant $u \in X$ according to

$$u \leftarrow x_0 \oplus F \odot (x_1 \ominus x_2). \quad (4)$$

A PSO particle is formed by its current position x , velocity v , personal and social best positions p and g . All these particle's properties can be encoded by using group elements, i.e., $x, v, p, g \in X$. Hence, given also the three scalar parameters $w, c_1, c_2 \geq 0$, the particle's new velocity $v' \in X$ and position $x' \in X$ are computed according to

$$v' \leftarrow [w \odot v] \oplus [(r_1 c_1) \odot (p \ominus x)] \oplus [(r_2 c_2) \odot (g \ominus x)], \quad (5)$$

$$x' \leftarrow x \oplus v', \quad (6)$$

where r_1, r_2 are two randomly generated numbers in $[0, 1]$.

In FA, the i -th computational firefly updates its position $x_i \in X$ by moving towards the positions of the brighter fireflies j_1, \dots, j_k , and by considering fitness as brightness. Formally, the new position x'_i is computed according to

$$x'_i \leftarrow x_i \oplus \bigoplus_{h=1}^k [\beta_0 \exp(-\gamma d(x_i, x_{j_h})^2) \odot (x_{j_h} \ominus x_i) \oplus (\alpha \odot \epsilon)], \quad (7)$$

where $\alpha, \beta_0, \gamma \geq 0$ are the FA scalar parameters, d is the distance induced by the finitely generated group at hand, and ϵ is a randomly generated discrete solution. Note how, with respect to the previous case, the FA update rule makes an explicit use of the discrete distance function induced by the finitely generated group at hand.

Generally, when the group is not Abelian, the composition is not independent of the terms ordering. This issue has been addressed in [3]. Finally note that many other evolutionary algorithms for numerical optimization can be adapted for combinatorial search spaces using our framework. Some examples are: artificial bee colony [7], bacterial foraging optimization [5], cuckoo search [17], and the fireworks algorithm [16].

4 A Formal Language Perspective

A formal language perspective on the algebraic framework described in Sect. 2 can be introduced by restricting our focus to a sub-class of finitely generated groups, namely, the finitely presented groups. All the concrete groups discussed in Sect. 2.4 are finitely presented. Moreover, other popular groups, like for example the braid group [6], are usually directly threatened by means of their presentation. Hence, restricting to finitely presented groups does not result in any practical issue for our purposes.

Formally, the group (X, \star) is finitely presented if there exists a presentation (H, R) such that: (i) $H \subseteq X$ generates X , i.e., (X, \star) is finitely generated by the generating set H ; (ii) R is a finite set of equivalence relations (made using the group operation \star) among the generators in H .

Interestingly, the presentation (H, R) of the group (X, \star) allows to interpret:

1. the generators in H as a set of symbols, i.e., an alphabet,
2. the elements in X as strings over the alphabet H , i.e., $x \in X$ if and only if $x \in H^*$,
3. the group operation \star as a concatenation of strings, i.e., given $x, y \in H^*$ then $x \star y = xy$, where xy denotes the concatenation of x and y ; and
4. the equivalence relations in R as rewriting rules for equivalent strings, i.e., if $(v, w) \in R$ then $vxw = vxv = wxv = wxw \in H^*$.

This formal language perspective allows to introduce a further level of generalization in our framework. Practically, we can facilitate the work of the algorithm designer by avoiding him/her to peak up a solutions' representation for the problem at hand. Indeed, the goal of this section is to show how a generic group presentation can be used to automatically derive generic implementations of three operators \oplus, \ominus, \odot .

The main idea is to encode the group elements (i.e., the solutions of the problem at hand) by means of their representation as strings of generators. Then, \oplus, \ominus, \odot can be automatically derived by simply introducing general procedures for element's inversion, composition and random decomposition that work directly on the string representation of the element.

By considering a generating set closed for inversions, i.e., $h \in H$ if and only if $h^{-1} \in H$, the inversion can be straightforwardly derived by exploiting the basic group properties. Formally, given a generic string $h_{i_1} h_{i_2} \dots h_{i_l} \in H^*$, its inverse is defined as

$$INV(h_{i_1} h_{i_2} \dots h_{i_l}) := h_{i_l}^{-1} \dots h_{i_2}^{-1} h_{i_1}^{-1}. \tag{8}$$

Composition, as already explained in the point 3 above, becomes a simple concatenation of strings. Formally, given $x, y \in H^*$:

$$CONCAT(x, y) := xy. \tag{9}$$

More interesting is the operation of random minimal decomposition *RandDec*. First note that a string representation of a group element is already

a decomposition in terms of generators, thus the problem becomes to find a procedure to simplify the string as much as possible. Intuitively, we can iteratively apply the equivalences in R in order to rewrite a string until it becomes of minimal length. The problem with this approach is that it is not easy to know when to stop. Luckily, the Knuth-Bendix (KB) completion algorithm [9] is a popular tool in computational algebra that allows to solve this issue.

KB takes in input the group presentation (H, R) and produces a terminating and confluent rewriting system RWS for the strings in H^* . RWS is nothing else than a set of rewriting rules such as $v \rightarrow w$ that can be iteratively applied to a string s until it does not match any rule in RWS , i.e., s has been reduced to its minimal length. Formally, we denote with $RWS(s)$ the minimal length string obtained by simplifying s with the rules in RWS . KB guarantees that the application of RWS terminates (i.e., it is a terminating system) and that the minimal form obtained for any given input string does not depend on the ordering by which the matching rules have been applied at every rewriting iteration (i.e., RWS is a confluent system). Therefore, KB can be run offline only once, because of that we can store the produced rewriting system RWS (it can be actually provided in different ways, one of them is as a finite state automaton) and execute it every time we need a minimal decomposition.

There is only one last issue. We need a randomized minimal decomposer, but the rewriting system produced by KB is confluent, i.e., deterministic. In order to introduce randomization, let consider that KB needs, as additional input, an arbitrary ordering (i.e., a permutation) of the generators in H . Hence, feeding KB with two different orderings on H produces two distinct rewriting systems RWS_1 and RWS_2 that both are terminating and confluent, but such that, in general, $RWS_1(s) \neq RWS_2(s)$. Therefore, in the offline computation stage, we can peak up k different permutations of H and run KB for k times in order to obtain k different rewriting system RWS_1, \dots, RWS_k . Then, a random minimal decomposition of a string $x \in H^*$ can be computed as

$$RandDec(x) := RWS_r(x), \quad (10)$$

where r is a random integer in $[1, k]$.

Summarizing, the (finite) presentation of a group allows to: (i) represent the group elements as string of generators, thus that no group (or problem) dependent encoding has to be considered, and (ii) provide general concrete implementations (i.e., working on any possible finitely presented group) of the element's inversion, composition and random minimal decomposition.

5 Automatic Algebraic Evolutionary Algorithms

Here we show how it is possible to automatically generate the implementation of an algebraic evolutionary algorithms by starting from a given group presentation.

By using the language-based tools provided in Sect. 4 we provide generic, but operative, implementations of the operators \oplus, \ominus, \odot for any possible group presentation.

Let $x, y \in H^*$ be two group elements represented as strings of generators. Then, addition and subtraction are defined according to, respectively,

$$x \oplus y := \text{CONCAT}(x, y), \quad (11)$$

and

$$x \ominus y := \text{CONCAT}(\text{INV}(y), x). \quad (12)$$

For the scalar multiplication \odot we provide language-based implementations of *Truncate* and *Extend* in, respectively, Figs. 3 and 4.

```

1: function TRUNCATE( $a \in [0, 1], x \in H^*$ )
2:    $x' \leftarrow \text{RandDec}(x)$   $\triangleright$  RandDec is defined in equation (10)
3:    $l \leftarrow \text{Length}(x')$ 
4:    $k \leftarrow \lceil a \cdot l \rceil$ 
5:    $z \leftarrow \epsilon$   $\triangleright \epsilon \in H^*$  is the empty string
6:   for  $i \leftarrow 1$  to  $k$  do
7:      $z \leftarrow \text{CONCAT}(z, x'_i)$   $\triangleright x'_i$  is the  $i$ -th generators of  $x'$ 
8:   end for
9:   return  $z$ 
10: end function

```

Fig. 3. Generic implementation of the truncation algorithm

```

1: function EXTEND( $a > 1, x \in H^*$ )
2:    $RWS \leftarrow$  a randomly chosen rewriting system from  $\{RWS_1, \dots, RWS_k\}$ 
3:    $x' \leftarrow RWS(x)$ 
4:    $l \leftarrow \text{Length}(x')$ 
5:    $k \leftarrow \lceil a \cdot l \rceil$ 
6:    $z \leftarrow x'$ 
7:   for  $i \leftarrow 1$  to  $k - l$  do
8:      $h \leftarrow$  a randomly generator such that  $\text{len}(RWS(zh)) = \text{len}(RWS(z)) + 1$ 
9:      $z \leftarrow \text{CONCAT}(z, h)$ 
10:  end for
11:  return  $z$ 
12: end function

```

Fig. 4. Generic implementation of the extension algorithm

Moreover, note that some algebraic evolutionary operators also needs the computation of the group distance $d(x, y)$ (see for example Eq. (7)). However, it is easy to show that $d(x, y) = \text{Length}(\text{RandDec}(x \ominus y))$.

Therefore, it is now straightforward to show how, using the language-based implementations of \oplus, \ominus, \odot , we can automatically derive an algorithm implementation by simply providing a group presentation and choosing the preferred

algebraic algorithmic schemes (e.g., the algebraic PSO described by Eqs. (5) and (6)). Figure 5 depicts and summarizes the main idea of this approach. Given a combinatorial problem to solve, the algorithm designer does not need to choose a solutions encoding. Indeed, he/she only needs to: provide a fitness function, choose its preferred algorithmic schemes (that uses algebraic operators), and provide a group presentation for the problem at hand. Note that the last step is usually straightforward, since group presentations for the set of problem solution is often directly available. The group presentation is then used in an offline computational stage where KB algorithm generates the rewriting systems RWS_1, \dots, RWS_k . Then, both the group presentation and the generated rewriting systems are used by the general implementations of \oplus, \ominus, \odot that, in turn, allow to obtain the desired evolutionary behavior. In conclusion, the proposed automatic mechanism substantially reduces the work of the algorithm's designer.

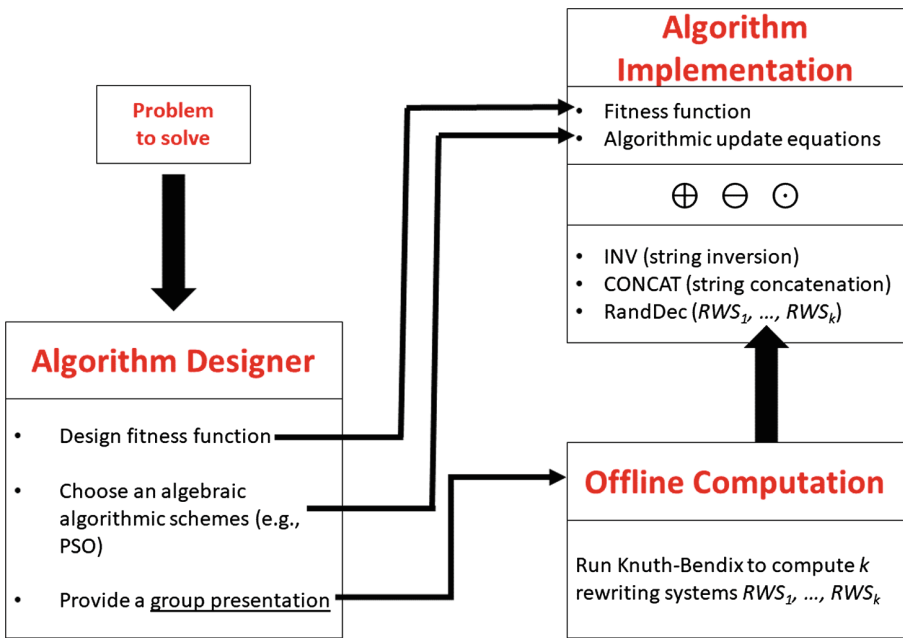


Fig. 5. Automatic generation of an algebraic evolutionary algorithm

6 Conclusion and Future Work

Starting from the algebraic framework for combinatorial optimization previously proposed in [1, 4, 11, 12], in this paper we have provided a mechanism to automatically derive concrete implementations of the framework for any search space representable by a finitely presented group.

To achieve this goal, a formal language perspective on the search space has been introduced. The main algebraic tool employed is the well known Knuth-Bendix completion algorithm.

As a future line of research we will consider the implementation of our proposal to derive algebraic evolutionary algorithms in order to address braid optimization problems [6] that have applications in the field of quantum computing, see for example [10].

References

1. Baioletti, M., Milani, A., Santucci, V.: Algebraic particle swarm optimization for the permutations search space. In: Proceedings of IEEE Congress on Evolutionary Computation, CEC 2017, pp. 1587–1594 (2017). <https://doi.org/10.1109/CEC.2017.7969492>
2. Baioletti, M., Milani, A., Santucci, V.: Linear ordering optimization with a combinatorial differential evolution. In: Proceedings of 2015 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2015, pp. 2135–2140 (2015). <https://doi.org/10.1109/SMC.2015.373>
3. Baioletti, M., Milani, A., Santucci, V.: An extension of algebraic differential evolution for the linear ordering problem with cumulative costs. In: Handl, J., Hart, E., Lewis, P., López-Ibáñez, M., Ochoa, G., Paechter, B. (eds.) PPSN 2016. LNCS, vol. 9921, pp. 123–133. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45823-6_12
4. Baioletti, M., Milani, A., Santucci, V.: A new precedence-based ant colony optimization for permutation problems. In: Shi, Y., et al. (eds.) SEAL 2017. LNCS, vol. 10593, pp. 960–971. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68759-9_79
5. Das, S., Biswas, A., Dasgupta, S., Abraham, A.: Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. In: Abraham, A., Hassanien, A.E., Siarry, P., Engelbrecht, A. (eds.) Foundations of Computational Intelligence Volume 3. SCI, vol. 203, pp. 23–55. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01085-9_2
6. Garside, F.A.: The braid group and other groups. *Q. J. Math.* **20**(1), 235–254 (1969)
7. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **39**(3), 459–471 (2007)
8. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
9. Knuth, D.E.: The genesis of attribute grammars. In: Deransart, P., Jourdan, M. (eds.) Attribute Grammars and their Applications. LNCS, vol. 461, pp. 1–12. Springer, Heidelberg (1990). https://doi.org/10.1007/3-540-53101-7_1
10. McDonald, R.B., Katzgraber, H.G.: Genetic braid optimization: a heuristic approach to compute quasiparticle braids. *Phys. Rev. B* **87**(5), 054414 (2013)
11. Santucci, V., Baioletti, M., Milani, A.: Algebraic differential evolution algorithm for the permutation flowshop scheduling problem with total flowtime criterion. *IEEE Trans. Evol. Comput.* **20**(5), 682–694 (2016). <https://doi.org/10.1109/TEVC.2015.2507785>

12. Santucci, V., Baiocchi, M., Milani, A.: Solving permutation flowshop scheduling problems with a discrete differential evolution algorithm. *AI Commun.* **29**(2), 269–286 (2016). <https://doi.org/10.3233/AIC-150695>
13. Santucci, V., Baiocchi, M., Milani, A.: A differential evolution algorithm for the permutation flowshop scheduling problem with total flow time criterion. In: Bartz-Beielstein, T., Branke, J., Filipič, B., Smith, J. (eds.) *PPSN 2014*. LNCS, vol. 8672, pp. 161–170. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10762-2_16
14. Schiavinotto, T., Stützle, T.: A review of metrics on permutations for search landscape analysis. *Comput. Oper. Res.* **34**(10), 3143–3153 (2007)
15. Storn, R., Price, K.: Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **11**(4), 341–359 (1997)
16. Tan, Y., Zhu, Y.: Fireworks algorithm for optimization. In: Tan, Y., Shi, Y., Tan, K.C. (eds.) *ICSI 2010*. LNCS, vol. 6145, pp. 355–364. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13495-1_44
17. Yang, X.S., Deb, S.: Cuckoo search via Levy flights. In: 2009 World Congress on Nature Biologically Inspired Computing (NaBIC), pp. 210–214 (2009)
18. Yang, X.-S.: Firefly algorithms for multimodal optimization. In: Watanabe, O., Zeugmann, T. (eds.) *SAGA 2009*. LNCS, vol. 5792, pp. 169–178. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04944-6_14



Multi-objective Optimization in High-Dimensional Molecular Systems

Debora Slanzi^{1,2}(✉), Valentina Mameli¹, Marina Khoroshiltseva¹,
and Irene Poli^{1,2}

¹ European Centre for Living Technology, S. Marco 2940, 30124 Venice, Italy
{debora.slanzi, valentina.mameli, marina.khoroshiltseva, irenepoli}@unive.it

² Department of Environmental Sciences, Informatics and Statistics,
Ca' Foscari University of Venice, Cannaregio 873, 30121 Venice, Italy

Abstract. The paper proposes a methodological approach to design complex experiments for multi-objective optimization. The strategy is based on evolutionary statistical inference to search for the optimal values in high-dimensional experimental spaces. We developed this approach to study a particular molecular system and discover the best molecules to be proposed as candidate drugs.

Keywords: Multi-objective optimization · Evolutionary strategies
Statistical models

1 Introduction

In the complex process of developing a new drug a major challenge concerns the construction of small molecules that interacting with a pharmacological target of interest can have a therapeutic effect on a particular disease. Current drug design practices involve the screening of large chemical libraries, composed of thousands or millions of compounds, with the aim of identifying a candidate molecule with suitable characteristics, known as a lead molecule. This lead molecule may not fulfil the properties needed to become a drug, such as Absorption, Distribution, Metabolism and Excretion (ADME) [1]. To achieve these properties, while retaining the interaction capacity with the target protein, the molecule must be modified for optimizing a set of variables. A simultaneous optimization is then required and the problem is framed as a multi-objective optimization problem with several conflicting objectives; see for example [2,3]. This field of research in drug discovery has been developed using different approaches, mainly based on the evolutionary principle. Among the most relevant contributions we mention the studies in *De novo Designs* [4], in *Molecular docking* [5] and in *Quantitative structure-activity relationships* [6]. These approaches have been successful in detecting the relevant information for discovering the molecule optimal values using computer based algorithms.

The optimization problem that we address in this study is concerned with a molecular system where each molecule is described by a very large number of features determining the high-dimensionality of the system. In order to discover the optimal molecular structure we intend to develop an evolutionary approach based on statistical models constructed to extract information in high-dimensional systems. In particular, we consider the Lasso model [7], Neural Networks [8], Stepwise Regression and Boosting models [9]. The main purpose of this paper is to develop an efficient approach that is able to find the best candidate molecules, testing a very small number of experimental compositions. This approach is then built on both the evolutionary paradigm and statistical models for high-dimensional experimental spaces. The paper extends the Evolutionary data Design for Optimization (*EDO*) proposed in [10, 11] allowing several objective functions to be optimized simultaneously. The approach, called *m-EDO* (multi-objective Evolutionary data Design for Optimization), drives the evolution towards the target by estimating and combining predictions from different models and different objective functions. We evaluate the method on the molecular library provided by [12] as a test set, investigated by [11, 13] and more recently by [14, 15].

The structure of the paper is as follows. In Sect. 2, we describe the main aspects of a multi-objective optimization problem. In Sect. 3, we briefly introduce the Model-Based Evolutionary Design for Optimization and the statistical models proposed for high-dimensional experimental spaces. In Sect. 4, we present the results achieved with the procedure for the data set provided by [12] and in Sect. 5, we present some concluding remarks.

2 The Multi-objective Optimization Problem

Discovering optimal values in high-dimensional systems can be a very difficult problem, in particular when the number of experimental tests (or observations) is small. Moreover, the optimal values can involve different properties of the system elements, introducing multiple (and possible conflicting) objective functions to be optimized simultaneously. This framing of the problem can make the search of the optimal values pretty hard.

In general, a multi-objective optimization problem can be described in the following way:

Consider a vector valued objective function $f : C \rightarrow \mathbb{R}^k$ with $C \subseteq \mathbb{R}^d$, where d is the dimension of each element of C and $f(c) = (f_1(c), \dots, f_k(c))$; search the element $c_0 \in C$ such that $f(c_0) \leq f(c)$ for all $c \in C$ (minimization) or such that $f(c_0) \geq f(c)$ for all $c \in C$ (maximization).

Frequently, in multi-objective optimization problems, there does not exist a solution, c_0 , which minimizes (or maximizes) all objective functions simultaneously. Therefore, the goal is to identify the Pareto optimal solutions which are the solutions that cannot be improved in any of the objectives without degrading at least one of the other objectives [2, 3].

Formally in a minimization problem, a point $c^* \in C$ is a Pareto optimal solution if for every $c \in C$ and $I = \{1, 2, \dots, k\}$ either,

$$\forall_{i \in I} f_i(c) = f_i(c^*) \quad (2.1)$$

or, there is at least one $i \in I$ such that

$$f_i(c) > f_i(c^*). \quad (2.2)$$

The set of optimal points is called the Pareto optimal set \mathcal{P}^* and the Pareto front \mathcal{F}^* , composed of all the values of the function f at the optimal points, is defined as:

$$\mathcal{F}^* := \{f(c) = (f_1(c), \dots, f_k(c)) \mid c \in \mathcal{P}^*\}. \quad (2.3)$$

In this research we will introduce a methodological approach to address multi-objective optimization in the context above described and related to a molecular system of interest for drug discovery. In particular, we propose a strategy which may aid the process of lead molecule optimisation.

3 Methods

3.1 The Model-Based Evolutionary Design for Optimization

In the drug discovery research field evolutionary algorithms can provide efficient and effective experimental designs; see for instance [16] and references therein. These evolutionary procedures, as described in [10, 11], enable to explore the whole experimental space meanwhile exploiting the capacity of statistical models to uncover relevant information. The evolutionary design for optimization, namely *EDO*-design, randomly generates a first initial population of experimental points. This initial set of experimental points is then tested in laboratory in order to derive the experimental response values. This data set (test composition and responses) are then used to build a class of statistical predictive models. The information gathered from these models is then used to drive the evolution towards the optimal value. In fact, with this information we can identify the next generation of experimental points which evolves from one generation to the next. The process continues until a pre-defined amount of objective-function evaluations is conducted. The *EDO* enables to capture the characteristics of the data and discover the optimum value by testing a very small set of points. This method has been developed for both single and multi-objective optimization.

Multi-objective optimization is based on the idea of guiding the evolution towards the target of the experimentation by building predictive statistical models for the different objective functions and using a linear combination of the best predicted values. The weights in the linear combination can be decided *a priori* with respect to the relevance of the objective functions. In order to distinguish between the single and multi-objective optimization, we indicate just by *EDO* the Evolutionary data Design for single Optimization and by *m-EDO* the Evolutionary data Design for multi-objective Optimization.

3.2 Models for Prediction

The Evolutionary data Design for Optimization, drives the evolution towards the target by estimating and combining predictions with different stochastic models for high dimensional settings, namely Lasso Regression, Stepwise Regression, Boosting, Neural Networks; see [15] and references therein. Herein, we briefly survey some features of these statistical tools.

Modelling data with a multiple linear regression we can write

$$y_i = X_i\beta + \epsilon_i, \quad i = 1, \dots, n \quad (3.1)$$

where y_i is the response variable, $X_i = (x_{i1}, \dots, x_{ip})$ is a p -dimensional vector of predictors (or covariates), $\beta = (\beta_1, \dots, \beta_p)^T$ is the regression vector of p unknown parameters, ϵ_i is the error term, and n is the number of observations. When the number of predictors is much larger than the number of observations ($p \gg n$) estimating regression models can be a very hard task. Under this setup, penalized regression procedures offer powerful methods to simultaneously estimate models and perform variable selection. Among these procedures, the most common is the least absolute shrinkage selection operator (Lasso); see [7].

According to Lasso model we estimate the parameter vector β by minimizing the following Lagrangian objective function

$$Q(\beta) = \frac{1}{2n} \sum_{i=1}^n (y_i - \sum_{j=1}^p \beta_j x_{ij})^2 + \lambda \sum_{j=1}^p |\beta_j|; \quad (3.2)$$

where λ is a tuning parameter which should be assessed by cross validation or information criteria (see [17]). The objective function in Eq. (3.2) is composed of two terms, the former is the least square loss function, the latter is the Lasso penalty, which imposes a constraint on the components of the vector β .

A different strategy for selecting the relevant variables in a regression framework are the step-wise selection methods. These are iterative techniques which allow to identify redundant variables by successively adding or removing variables on the basis of statistical significance criteria. Alternative to the linear regressions the Neural Networks (NN) models can be considered for this structure of data. Neural Networks, in fact, are suitable models for dealing with data characterized by complex non-linear relationships and have become a popular tool for many applications in a wide range of fields including drug discovery research [18]. From a statistical perspective of Neural Network models, we refer the reader to [8, 19]. The topology of a Neural Network can be described as a collection of nodes, namely neurons, which are arranged into ordered layers. A Neural Network usually contain input, hidden and output layers. Considering a Feed Forward NN the input layer communicates information to one or more hidden layers linked to the output layer of this net. With the single hidden layer, the dynamics of the information can be summarized by the following expression

$$y = f(\phi(X, w)) \quad (3.3)$$

where y is the output neuron which can be univariate or multivariate, X represents a set of covariates, w are the weight connections among the neurons directly connected. The function f is called the activation function whose form depends on the problem under consideration. This function can take different forms, which include the linear, the sigmoidal, the logistic or the Gaussian forms. Frequently, the function ϕ is linear and is called the propagation function, it represents the relationship among a neuron and their predecessors. There are several training algorithms for estimating Neural Networks parameters, such as the classical Back-propagation, the scaled conjugate gradient Back-propagation and the Bayesian regularization Back-propagation methods; see [20–22]. To analyse complex structure in the data also the family of boosting algorithms can be considered. Boosting is a class of ensemble techniques that construct multiple estimates or predictions by using combined and averaged estimates or predictions [9]. More formally, the aim of boosting algorithms is to construct or estimate a complex relationship between a set of covariates and a response variable, i.e. $y = f(X)$. This goal will be achieved by minimizing a loss function which measures the discrepancy among y and f by estimating M times a particular model by using weighted fitting and at the end of M iterations we consider a weighted sum of the estimate founds. The specificity of the algorithms is related to different loss functions which depend on the characteristics of the response variable. In our approach we consider the least square loss function as in Lasso framework for estimating a linear regression model; see [9] for an introduction on the boosting algorithm from a statistical perspective.

4 Lead Optimization in a Molecular System

4.1 Data Description

In this research we address the lead optimization of MMP-12 Inhibitors, using the library of molecules made publicly available by [12]. This library consists of 2500 molecules described by the presence of 22272 fragments. In our approach fragments take the role of predictors and are represented as binary variables indicating the presence or absence of each fragment into the molecule. The analysis of these data showed the presence of linear dependence among the predictors (fragments) leading then to a reduction of the total amount of fragments to 4059. Given the high-dimensionality of the system for the very high number of fragments that characterizes each molecule, we adopted the Formal Concept Analysis and reduced the number of fragments to 175; see [14]. Each molecule is then characterized by a set of properties: the pharmacological activity at the target protein; physicochemical properties, such as the solubility; the toxicity property; and structural properties, such as the lipophilicity and the molecular weight. In particular, the pharmacological activity at the target protein, defined as the capacity to produce physiological or chemical effects by the binding of a compound to the therapeutic target, will be denoted by *Activity*. The solubility of a compound is the capacity to dissolve in a liquid and it will be denoted by

Solubility. The toxicity refers to the capacity of any chemicals to produce undesirable effects, and it will be called *Safety*. The lipophilicity is the capacity of the compound to dissolve into lipid structures, and it will be denoted by *cLogP*. The molecular weight relates to the compound size and will be denoted by *MW*. For a more detailed explanation of these biological concepts we refer to the book [1], which describes the influence of each of compound property on ADME and toxicity. Therefore, for this system the experimental response variables (molecular properties) that we considered are: *Activity*; *Solubility*; *Safety*; *cLogP* and *MW*. These response variables represent the target of our optimization study, and some summary statistics of the molecular library made available are reported in the following:

- *Activity*, y_1 : the maximum value is **8**, which corresponds to the optimal value. The **99-th percentile** of the response variable distribution is **7.5**. The target is the **maximization** of y_1 .
- *Solubility*, y_2 : the maximum value is **-1.766**, which corresponds to the optimal value. The **99-th percentile** of the response variable distribution is **-2.415**. The target is the **maximization** of y_2 .
- *Safety*, y_3 : the maximum value is **3.6262**, which corresponds to the optimal value. The **99-th percentile** of the response variable distribution is **3.2309**. The target is the **maximization** of y_3 .
- *cLogP*, y_4 : the minimum value is **-2.505**, which corresponds to the optimal value. The **1-th percentile** of the response variable distribution is **0.033**. The target is the **minimization** of y_4 .
- *MW*, y_5 : the minimum value is **291.3**, which corresponds to the optimal value. The **1-th percentile** of the response variable distribution is **339.3**. The target is the **minimization** of y_5 .

Figure 1 depicts the box-plot of the distribution of the five properties that characterize the set of 2500 molecule, and the blue stars represent the optimum value of each response variable.

The aim of this study is to develop a multi-objective optimization procedure based on experimental data (no simulation), and involving a very small number of experimental tests, to avoid unnecessary waste of research resources. Knowing the whole experimental space (complete library) allowed us to evaluate the performance of the approach in searching the best response values repeating the procedure 1000 times. In order to obtain drug candidates with suitable properties, some constraints are imposed on the molecular properties. In particular, we consider molecules with: *Activity* values $y_1 > 6$, *Solubility* values $y_2 > -3$, *Safety* values $y_3 > 2.57$, *cLogP* values $y_4 < 3$ and *MW* values $y_5 < 450$. Then the goal of the multi-objective optimization is to discover the molecules (three molecules in the library) that satisfy the constraints of the problem and reach their best response values. These molecules are described (in red) in Fig. 2, and represent the molecules belonging to the Pareto Front of *Solubility* and *Safety* with the constraints above described on the other properties. Moreover, the dashed lines represent the constraints values on *Solubility* and *Safety*, respectively. The response values of the three best molecules, goal of our study, are

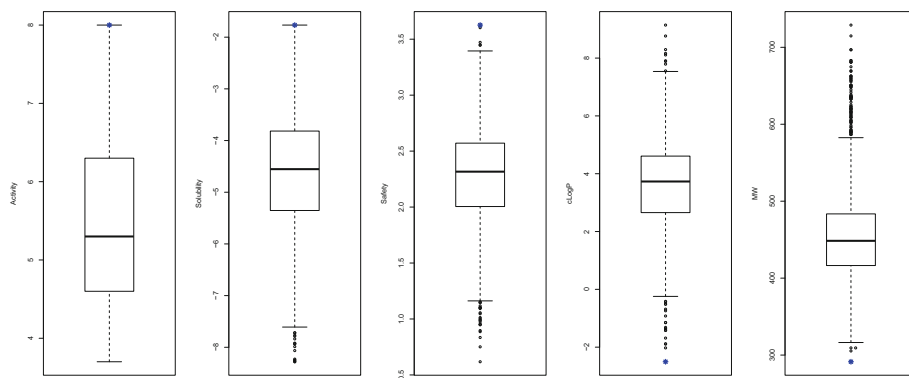


Fig. 1. Box-plot of the response values distribution, from left to right *Activity*, *Solubility*, *Safety*, *cLogP*, and *MW*. (Color figure online)

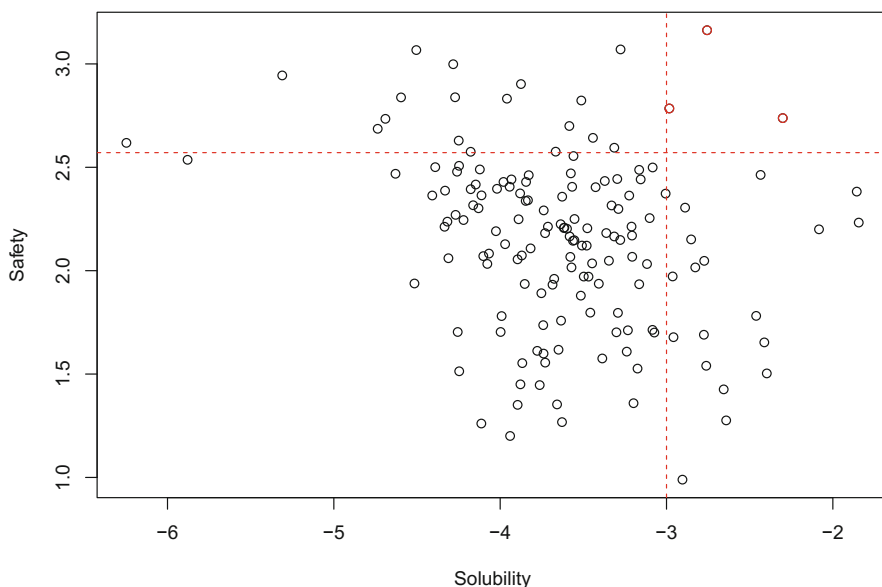


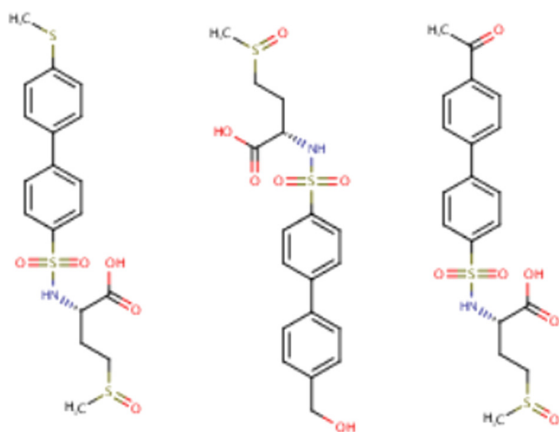
Fig. 2. The molecule values of *Solubility* and *Safety*, respecting the defined constraints for *Activity*, *cLogP*, and *MW* and in red the molecules belonging to the Pareto Front. The dashed lines represent the constraints values on *Solubility* and *Safety*, respectively. (Color figure online)

presented in Table 1. In this study we would like to discover these three molecules by conducting the minimum possible set of experimental tests.

The chemical representation of these molecules as reported in Fig.3 has been obtained by using the **SwissADME** web tool freely available at <http://www.swissadme.ch/>; see [23].

Table 1. Values of the five response variables assumed by the three best molecules.

	<i>Activity</i>	<i>Solubility</i>	<i>Safety</i>	<i>cLogP</i>	<i>MW</i>
Molecule 1	6.90	-2.98	2.78	1.39	427.56
Molecule 2	6.20	-2.76	3.16	0.81	425.48
Molecule 3	6.50	-2.30	2.74	0.38	423.50

**Fig. 3.** Chemical structure of the three target molecules.

4.2 The Evolutionary Procedure and the Optimization Results

At each generation the population size of the evolutionary algorithm is 20 and the algorithm is run for 7 generations, with the constraint that all individuals tested must be different. In this study the maximum number of tests considered is 140 on the total of 2500 candidate compositions. The process is iteratively repeated, generation after generation, maintaining the same size in each population of experimental points and ends when the maximum total number of experimental points is reached. The structure of evolutionary approach consists in randomly selecting an initial small population, in this study 20 molecules, and then evaluating the response variables values of each molecule. In the evolutionary algorithm, the next population of experiments is then built by selecting the 20 molecules with the best predicted response values.

At first, to better understand the performance of the approach, we developed the procedure of single objective optimization for each response variable. The evolution has been driven by the information achieved with the Lasso model, Stepwise regression, Boosting, Neural Networks, and the mixture of these three linear models (hereafter Mixture of linear Models) [14]. The architecture of the Neural Network used for this single-objective problem consists of 175 input neurons, one hidden layer with 7 neurons and one output layer with one neuron.

The activation function is the sigmoidal function, and the Neural Network has been fitted by using Bayesian regularization Back propagation; see [22].

We study the robustness of the procedure with respect to the change of the initial population by repeating the entire process 1000 times. The good performance of the procedure is evaluated in its capacity to uncover the optimum value and the set of values in the region of optimality (the 1% best values of the distribution), conducting a very limited set of experimental tests (5.6% of the whole experimental space). The results achieved for the single optimization process are represented in Table 2.

Table 2. Single objective optimization: number of runs (out of 1000 runs) in which *EDO* uncovers the optimum value and values in the region of optimality (1% best values of the distribution).

Objective		Lasso	Stepwise	Boosting	Mixture of linear models	NN
<i>Activity</i>	Optimum	844	782	665	916	660
	Region of optimality	1000	995	998	1000	990
<i>Solubility</i>	Optimum	875	745	872	912	556
	Region of optimality	995	998	1000	1000	996
<i>Safety</i>	Optimum	387	358	278	467	228
	Region of optimality	1000	1000	1000	1000	999
<i>cLogP</i>	Optimum	848	821	917	918	760
	Region of optimality	950	946	981	1000	945
<i>MW</i>	Optimum	738	822	751	887	346
	Region of optimality	905	966	956	1000	780

From these results we can learn that *EDO* procedure is able to achieve the best response values in a very high proportion of the 1000 runs, showing also a better performance of the Mixture of linear Models with respect to the single model optimization. Concerning the response values in the region of optimality (1% best values of the distribution) we observe that the Mixture of linear Models is able to achieve these values in all the 1000 runs and for all the variables.

We then address the problem of the multi-objective optimization by extending the *EDO* approach which involves the evolution driven by the information achieved with the Lasso model, Neural Network, and the Mixture of the Lasso and the Neural Network models [14]. The architecture of the Neural Network in the multi-objective optimization has the same topology proposed for the single-objective optimization except that the output layer consists of 3 neurons. In fact, we consider 3 response variables *Activity*, *Solubility*, *Safety*. The variables *cLogP* and *MW* are not taken into consideration in the multi-objective procedure because the *cLogP* is highly correlated with the *Solubility*, and the *MW* could be easily predicted on the basis of its amino acids composition; see [24]. In particular for the multi-objective procedure, we selected in a random way 20

Table 3. Multi-objective optimization: number of runs (out of 1000 runs) in which *m*-EDO uncovers the best molecules.

Number of best molecules	Lasso	NN	Mixture of models
0	130	161	92
1	43	59	51
2	320	288	384
3	507	492	473
At least one	870	839	908

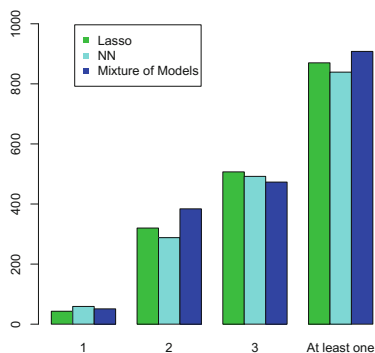


Fig. 4. Multi-objective optimization: best molecules found in 1000 runs.

experimental points from the whole population of 2500 molecules, then we built the model on these 20 data and predict the response variables values by using an estimated model, as in the single optimization procedure. We then associated a weight at each objective value and derived the linear combination of objectives. The molecules with the best estimated linear combination of the objective values are then selected for the next generation of experimental points. In the following table (Table 3) we present the results for the multi-objective optimization achieved with the Lasso model, the Neural Network model (NN) and the Mixture of Lasso and Neural Network models (Mixture of Models) and in Fig. 4 we depict these results. The three ways chosen to optimize give similar results in discovering the three best molecules. We notice that Mixture of models outperforms the alternatives in discovering at least one molecule of the three in more than 90% of 1000 runs.

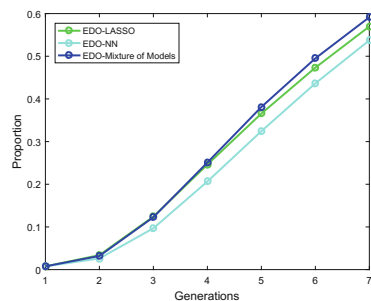
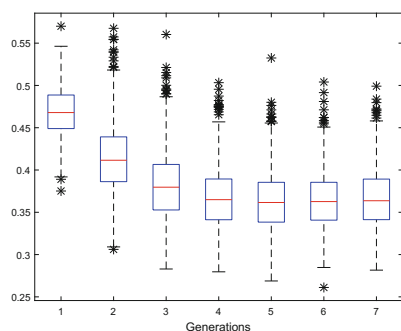


Fig. 5. Evolution through generations. Left: box-plot of the mean of the results achieved in 1000 runs by using the Mixture of Models. Right: proportion (average on 1000 runs) of the results found in the 1% best values of the distribution.

From the generation results, as presented in Fig. 5, we notice the relevance of the evolutionary principle in the search process: from the first generation there is a clear tendency of the procedure to converge towards the optimal value. From the left-hand panel of Fig. 5, we can notice that the mean of the objective response value is decreasing in each generation and get closer to the optimal solution which is identified by the value 0. In fact, we transform our variable values to lie in the interval $[0, 1]$ and the multi-objective optimization becomes just a minimization of the linear combination of the objective response values. Moreover, we also notice in Fig. 5 that the median of the distribution decreases generation after generation and becomes stable after the fourth generation. From the right-hand panel of Fig. 5, we can notice that the proportion of the new objective response values, i.e. the linear combination of the objective response values, found in the region of optimality (1% best values of the distribution) increases generation after generation.

5 Concluding Remarks

The purpose of this research was the development of a methodological strategy able to address the multi-objective optimization problem for complex experimentation conducting a very small number of tests. The procedure proposed is a model-based evolutionary strategy for designing experiments, which involves the construction and the estimation of predictive linear and non-linear models. The study of a particular molecular system for drug discovery problems shows the very good performance of the approach that we propose. Moreover, we would like to stress that we achieve these results by conducting an extremely small number of generations (7 generations) that usually is regarded too small to even approach convergence of the algorithm.

Acknowledgements. The authors would like to acknowledge the fruitful collaboration with Darren Green and his Molecular Design group at GlaxoSmithKline (GSK), Medicines Research Centre, Stevenage (UK).

References

1. Kerns, E.H., Di, L.: *Drug-Like Properties: Concepts, Structure Design and Methods: From ADME to Toxicity Optimization*. Academic Press, San Diego (2008)
2. Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: *Evolutionary Algorithms for Solving Multi-Objective Problems*. Genetic and Evolutionary Computation. Springer, New York (2006). <https://doi.org/10.1007/978-0-387-36797-2>
3. Lobato, F.S., Steffen, V.: *Multi-objective Optimization Problem*. SpringerBriefs in Mathematics. Springer International Publishing, Cham (2017). <https://doi.org/10.1007/978-3-319-58565-9>
4. Ekins, S., et al.: Evolving molecules using multi-objective optimization: applying to ADME/Tox. *Drug Discov. Today* **15**, 410–451 (2010)
5. Li, H., et al.: An effective docking strategy for virtual screening based on multi-objective optimization algorithm. *BMC Bioinform.* **10**(58), 1–12 (2009)

6. Soto, A.J., et al.: Multi-objective feature selection in QSAR using a machine learning approach. *QSAR Comb. Sci.* **28**, 1509–1523 (2009)
7. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **58**(1), 267–288 (1996)
8. Cheng, B., Titterton, D.M.: Neural networks: a review from a statistical perspective. *Stat. Sci.* **9**(1), 2–30 (1994)
9. Bühlmann, P., Hothorn, T.: Boosting algorithms: regularization, prediction and model fitting. *Stat. Sci.* **22**(4), 477–505 (2007)
10. Baragona, R., Battaglia, F., Poli, I.: *Evolutionary Statistical Procedures: An Evolutionary Computation Approach to Statistical Procedures Designs and Applications*. Springer, Heidelberg (2013). <https://doi.org/10.1007/978-3-642-16218-3>
11. Borrotti, M., De March, D., Slanzi, D., Poli, I.: Designing lead optimization of MMP-12 inhibitors. *Comput. Math. Methods Med.* **2014**, 8 (2014). Article ID 258627
12. Pickett, S.D., Green, D.V.S., Hunt, D.L., Pardoe, D.A., Hughes, I.: Automated lead optimization of MMP-12 inhibitors using a genetic algorithm. *ACS Med. Chem. Lett.* **2**(1), 28–33 (2011)
13. Brown, P.J., Ridout, M.S.: Level-screening designs for factors with many levels. *Ann. Appl. Stat.* **10**(2), 864–883 (2016)
14. Giovannelli, A., Slanzi, D., Khoroshiltseva, M., Poli, I.: Model-based lead molecule design. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016*. CCIS, vol. 708, pp. 103–113. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_9
15. Mameli, V., Lunardon, N., Khoroshiltseva, M., Slanzi, D., Poli, I.: Reducing dimensionality in molecular systems: a Bayesian non-parametric approach. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016*. CCIS, vol. 708, pp. 114–125. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_10
16. Lameijer, E.-W., Bäck, T., Kok, J.N., Ijzerman, A.D.P.: Evolutionary algorithms in drug design. *Nat. Comput.* **4**(3), 177–243 (2005)
17. Akaike, H.: Information theory and an extension of the maximum likelihood principle. In: Petrov, B.N., Csaki, F. (eds.) *2nd International Symposium on Information Theory*, pp. 267–281. Akademiai Kiado, Budapest (1973)
18. Livingstone, D.: *Artificial Neural Networks: Methods and Applications*. Humana Press, Totowa (2008)
19. Poli, I., Jones, R.D.: A neural net model for prediction. *J. Am. Stat. Assoc.* **89**(425), 117–121 (1994)
20. Foresee, F.D., Hagan, M.T.: Gauss-Newton approximation to Bayesian learning. In: *Proceedings of the International Joint Conference on Neural Networks* (1997)
21. Moller, M.F.: A scaled conjugate gradient algorithm for fast supervised learning. *Neural Netw.* **6**, 525–533 (1993)
22. MacKay, D.J.C.: Bayesian interpolation. *Neural Comput.* **4**(3), 415–447 (1992)
23. Daina, A., Michielin, O., Zoete, V.: SwissADME: a free web tool to evaluate pharmacokinetics, drug-likeness and medicinal chemistry friendliness of small molecules. *Sci. Rep.* **7**, Article No. 42717 (2017)
24. Guan, Y., Zhu, Q., Huang, D., Zhao, S., Jan, L., Peng, J.: An equation to estimate the difference between theoretically predicted and SDS PAGE-displayed molecular weights for an acidic peptide. *Sci. Rep.* **5**, Article No. 13370 (2015)



Multiple Network Motif Clustering with Genetic Algorithms

Clara Pizzuti^(✉) and Annalisa Socievole

National Research Council of Italy (CNR), Institute for High Performance Computing
and Networking (ICAR), Via Pietro Bucci, 7-11C, 87036 Rende (CS), Italy
{clara.pizzuti,annalisa.socievole}@icar.cnr.it

Abstract. The definition of community, usually, relies on the concept of edge density. Network motifs, however, have been recognized as fundamental building blocks of networks and, similarly to edges, may give insights for uncovering communities in complex networks. In this work, we propose a novel approach for identifying communities of network motifs. Differently from previous approaches, our method focuses on searching communities where nodes simultaneously participate in several types of motifs. Based on a genetic algorithm, the method finds a number of communities by minimizing the concept of multiple-motifs conductance. Simulations on a real-world network show that the proposed algorithm is able to better capture the real modular structure of the network, outperforming both motifs-based and classic community detection algorithms.

Keywords: Community detection · Network motifs
Evolutionary techniques · Genetic algorithm

1 Introduction

Complex networks contain small subgraphs named *network motifs* [11], which are pattern of interconnections recurring more frequently than expected in a random network. The frequency of a motif describes the number of times this motif appears within the network. High frequencies of certain motifs are possible due to the important functions they play in a network. For example, the *feed forward loop* and the *bifan* motifs shown in Fig. 1(a) and (d), respectively, have been found to be highly frequent into the genetic regulation networks of *E. coli* and *S. Cerevisiae*, as well as into the *C. elegans* neurons network. It is worth noting that multiple motifs usually coexist within a network. Figure 2 shows a subgraph of *Florida Bay food web* network [18], where different microorganisms interact through multiple motifs. We highlight here three types of network motifs: M_5 , M_6 (Fig. 1(b)) and M_8 (Fig. 1(c)). In motif M_5 , Water POC serves as energy source for Free Bacteria and Meroplankton, and Meroplankton for Free Bacteria. In motif M_6 , Free Bacteria acts also as energy source for Arcatia tonsa, and both nodes are served by Input. Finally, interaction patterns like Input serving Water flagellates and Water ciliates (motif M_8) occur many times.

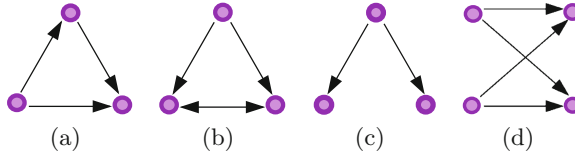


Fig. 1. (a) M_5 (feed-forward loop), (b) M_6 , (c) M_8 , and (d) M_{bifan} motifs.

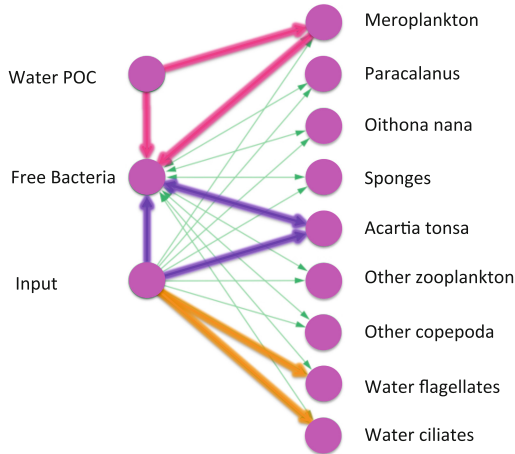


Fig. 2. Multiple motifs coexisting in a subgraph of Florida Bay food web network: M_5 , M_6 and M_8 . The edges composing these three motifs are highlighted in pink, purple and orange, respectively. (Color figure online)

Although network motifs have been recognized as “fundamental units of networks” [3], few studies explore the role these subpatterns have in community detection. Arenas et al. [1] show how motifs can be used to define a *motif-based modularity*, i.e. how motif-based modules present more motifs than a random division. Specifically, they extended the original definition of modularity introduced by Girvan and Newman [6] to deal with classes of motifs, and showed that the detected partitions are different with respect to those obtained by optimizing the classical modularity. A spectral method based on the generalized modularity [17] has been proposed by the same authors, and the differences between the obtained community structures on several networks are highlighted. In a recent work, Benson et al. [2] proposed a tensor spectral clustering method that clusters nodes according to the motif specified in input by the user. First, the higher-order structures involving multiple nodes are encoded by means of tensors (i.e., multidimensional matrices). Then, the method searches a partitioning that does not cut the motifs. Another work [3] by the same authors, described in detail in

the next section, extends the concept of *conductance* [16] to network motifs for finding cluster of motifs with low *motif conductance*.

One of the main drawbacks of the aforementioned approaches is that the number of communities must be fixed in advance. In a previously work [14], we proposed *MotifGA*, an evolutionary motifs-based algorithm for community detection using *Genetic Algorithms (GAs)* [7] and a type of motif as input for discovering a number of motif-based communities minimizing motif conductance. Here and in all the previous cited related works, motif-based clustering is thus performed fixing a type of motif and exploring the communities based only on that motif, without considering the coexistence of multiple motifs.

In this paper, we propose *M-MotifGA*, a genetic algorithm for detecting communities in complex networks simultaneously considering different motifs. The method evolves a population of individuals by minimizing the concept of *multiple-motifs conductance*, and finds a partition of the network into k communities, with k determined by the best local solution optimizing the multiple-motifs conductance as fitness function. A comparison with the approach of Benson et al. [3], with a variant of this approach we developed here for taking into account multiple motifs, and with the two best known community detection methods *Louvain* [4] and *Infomap* [15] shows that *M-MotifGA* obtains results better than those found by the other state-of-the-art methods.

The paper is organized as follows. Section 2 introduces the concepts of conductance when motifs are considered and defines the problem we tackle. Section 3 describes our method. Section 4 details the dataset used to perform our experiments and the results obtained. Finally, Sect. 5 concludes the paper.

2 Network Motif Clustering

In this section, we start recalling the concepts of *network motif*, *conductance*, *motif conductance* and *multiple-motifs conductance*. Then, we describe the method proposed by Benson et al. [3] and the introduction of multiple motifs within their method.

Given a graph $G = (V, E)$ with weights W , $n = |V|$ number of vertices, and $m = |E|$ number of edges, a *motif* M of G on r nodes $\{v_1, \dots, v_r\}$, represented by a sub-adjacency matrix of size $r \times r$, is defined as a subgraph of G presenting a particular pattern of interconnections. Figure 1 shows three types of motifs among three nodes (Fig. 1(a), (b), and (c)) and a motif involving four nodes (Fig. 1(d)). Their labeling follows the same convention adopted in [3].

Given the diagonal degree matrix D of G defined as $D_{ii} = \sum_{j=1}^n W_{ij}$, and a set $S \subset V$ of nodes, the *cut* of S , denoted $cut(S)$, is defined as the sum of edge weights having one endpoint in S and the other in $\bar{S} = V - S$:

$$cut(S) = \sum_{i \in S, j \in \bar{S}} W_{ij} \quad (1)$$

The *conductance* of S is defined as

$$\phi(S) = \frac{cut(S)}{\min(vol(S), vol(\bar{S}))} \tag{2}$$

where $vol(S) = \sum_{i \in S} D_{ii}$ is the weighted sum of edge end points in S .

By substituting an edge with a motif instance of M , the conductance of S can be generalized to motifs as follows

$$\phi_M(S) = \frac{cut_M(S)}{\min(vol_M(S), vol_M(\bar{S}))} \tag{3}$$

where $\phi_M(S)$ is defined *motif conductance*, $cut_M(S)$ is the number of motif instances of M with at least a node in S and another in \bar{S} , and $vol_M(S)$ is the number of instances of M contained in S .

Problem definition (single-motif). Fixed a network motif M , find a set of nodes S such that: (1) they participate in as many instances of M as possible, and (2) cutting instances of M is avoided, i.e. all the nodes of M should belong to either S or \bar{S} .

Benson et al. [3] proposed a method for partitioning V into the complementary sets S and \bar{S} that, given a motif M , minimizes the motif conductance $\phi_M(S)$. The method works on the *motif adjacency matrix* W_M , where each element represents the number of times two nodes appear in an instance of M . When there are nodes that do not participate in any motif, these nodes are discarded from W_M . Then, the eigenvector corresponding to the second smallest eigenvalue of the normalized motif Laplacian matrix is computed. The components of the eigenvector generate an ordering of nodes, which produces nested sets of nodes. The set of nodes with the smallest motif conductance is proven to be a near-optimal partition. Further details on the approach can be found in [3]. For obtaining a partition with more than two communities, the method, named *Motif Recursive bi-partitioning (MRbi-part)*, can be recursively executed on S and \bar{S} , until the desired number of clusters is obtained.

When considering M_1, M_2, \dots, M_q motifs simultaneously, the *multiple-motifs conductance* is defined as

$$\phi_{MM}(S) = \frac{\sum_{j=1}^q \alpha_j cut_{M_j}(S)}{\min(\sum_{j=1}^q \alpha_j vol_{M_j}(S), \sum_{j=1}^q \alpha_j vol_{M_j}(\bar{S}))} \tag{4}$$

where each $\alpha_j \geq 0$ gives a weight to the impact of motif M_j on the considered network.

Problem definition (multiple-motifs). Given a set of q network motifs M_1, M_2, \dots, M_q , find a set of nodes S such that (1) they simultaneously participate in as many instances of all the considered motifs as possible, and (2) cutting instances of any $M_j, j = 1, \dots, q$ are avoided.

In the next section, we propose to solve the problem of finding a division on a network based on multiple motifs by applying a Genetic Algorithm. Specifically,

the proposed algorithm minimizes the multiple-motifs conductance computed on the motif adjacency matrices of the single motifs, associated with the graph G representing the network. For comparing our method to another method based on multiple motifs, we also modified *MRbi-part* extending its code such that the multiple-motifs conductance is the measure to minimize. As such, we considered a *weighted motif adjacency matrix* $W_{Mw} = \sum_{j=1}^q \alpha_j W_{M_j}$ for running the method. We denominate this extension as *MRbi-part*_{MM}. It is worth noting that, differently from the methods by Benson et al., our method does not need a prior setting of the number of communities to find. This number is automatically provided by decoding the solution obtained by the method, i.e. the solution with the lowest local optimum value of conductance.

3 *M-MotifGA* Description

The algorithm we propose, named *M-MotifGA* is based on our previous work [14], where we proposed *MotifGA*, an approach to motif network clustering exploiting a genetic algorithm, that evolves a population of individuals by minimizing motif conductance. Similarly to *MotifGA*, *M-MotifGA* obtains the simultaneous partition of a network into k communities, with k determined by the best local solution optimizing the fitness function. However, differently from *MotifGA*, the fitness function used in *M-MotifGA* is the multiple-motifs conductance.

A GA-based method basically evolves a population of individuals initialized at random, and performs variation and selection operators to increase the value of a criterion function, while exploring the search space during the optimization process. *M-MotifGA* uses the locus-based adjacency representation [13] for representing the problem, uniform crossover and neighbor-based mutation for evolving individuals. In the locus-based representation, an individual I is represented as a vector of n genes. Each gene can assume a value j in the range $\{1, \dots, n\}$: when a value j is assigned to the i th node, nodes i and j are linked. A decoding step identifies all the connected components of the graph which correspond to the network division in communities. Uniform crossover generates a random binary mask of length equal to the number of nodes, and an offspring is obtained by selecting from the first parent the genes in the mask set to 0, and from the second parent the genes in the mask set to 1. Finally, the mutation operator randomly changes the value j of a gene to one of its neighbors.

M-MotifGA receives in input the graph $G = (V, E)$ and the set of motifs M_1, M_2, \dots, M_q , and performs the following steps:

1. compute the motif adjacency matrices $W_{M_1}, W_{M_2}, \dots, W_{M_q}$;
2. take the largest connected component $W_{M_j}^{max}$ of W_{M_j} for each motif M_j of the q motifs;
3. obtain the weighted graph $G_{M_j} = (V_{M_j}, E_{M_j})$ corresponding to $W_{M_j}^{max}$ for each $1 \leq j \leq q$;
4. compute the weighted graph $G_M = \sum_{j=1}^q G_{M_j}$;

5. run the GA on G_M for a number of iterations by using multiple-motifs conductance as fitness function to minimize, uniform crossover and neighbor mutation as variation operators;
6. obtain the partition $C = \{C_1, \dots, C_k\}$ corresponding to the solution with the lowest fitness value;
7. merge two communities if the number of inter-cluster connections is higher than the number of intra-cluster connections.

Note that the weighted graphs G_{M_j} associated with the largest connected components of the motif adjacency matrices may have different numbers of nodes. In this case, before Step 4, the algorithm computes the subset of nodes which are common to all the G_{M_j} graphs. Then, the matrices G_{M_j} will be reorganized in order to contain only the rows and the columns related to that subset of nodes.

In the next section we present the results obtained by our algorithm and compare them with those returned by state-of-the-art methods. Moreover, we also investigate a variant of our approach, named *MS-MotifGA*, that uses as fitness function the sum of the motif conductance of the single motifs, that is $\phi_{MS}(S) = \phi_{M_1}(S) + \phi_{M_2}(S) + \dots + \phi_{M_q}(S)$.

4 Experimental Evaluation

To validate our algorithm, we performed several simulations using Matlab 2015b and the Global Optimization Toolbox. Specifically, we compared our algorithm with other well-known state-of-the-art algorithms in terms of *NMI* [5], *ARI* [9] and *F1* [10] indexes. The results for *M-MotifGA* have been averaged over 10 runs of the algorithm, setting the population size to 100, the number of generations to 200, the mutation rate to 0.2, and the crossover rate to 0.8. These parameter values have been fixed by employing a trial-and-error procedure on the benchmark data set. Moreover, for computing the multiple-motifs conductance, we equally weighted all motifs using $\alpha_j = 1$. For *MRbi-part* we set to 4 the number of communities to find, as suggested by Benson et al. for this dataset, since a higher number of communities would give higher motif conductance values and, thus, worse results for their algorithm. Specifically, we applied the motif recursive bipartitioning method twice in order to obtain the desired number of communities. The following subsections describe the dataset and performance indexes used, and the algorithms taken into account for testing the effectiveness of *M-MotifGA*.

4.1 Dataset

We analyze the **Florida Bay food web** dataset containing the data of an ecosystem food web. Converting these data into a network graph, nodes can be considered organisms and species, and edges the directed carbon exchange between species. For clustering this network, we consider the motifs M_5 , M_6 and M_8 shown in Fig. 1. M_5 , considered a building block for food webs, describes the

hierarchical flow of energy between species i and j which are energy sources for species k , while i is an energy source for both. M_6 , on the contrary, models two species that exchange energy and compete to receive energy from a third specie. This motif has been shown to be prevalent within this network, resulting in a rich high-order modular structure. Finally, M_8 corresponds to a single specie feeding two non-interacting species.

The original Florida Bay food web network is composed by 128 nodes and 2106 edges. For detecting communities, we consider a subset of 62 nodes for which the ground truths are known. Specifically, two ground truths, denoted as $GT1$ and $GT2$, are available and are relative to the two large connected components resulting from the analysis of the adjacency matrix of motif M_6 . Table 1 shows the 50 nodes corresponding to the first component and the 12 nodes of the second component. The remaining 66 nodes are isolated. In $GT1$ nodes are classified into 11 different categories ('*demersal producer*', '*seagrass producer*', '*algae producer*', '*microbial microfauna*', '*zooplankton microfauna*', '*sediment organism microfauna*', '*macroinvertebrates*', '*pelagic fishes*', '*benthic fishes*', '*demersal fishes*', and '*detritus*'). In $GT2$, on the contrary, nodes are categorized into 7 groups: '*producer*', '*microfauna*', '*macroinvertebrates*', '*pelagic fishes*', '*benthic fishes*', '*demersal fishes*', and '*detritus*'. Basically, $GT2$ considers all the *producer* and *microfauna* subcategories of $GT1$ as unique macro categories.

The largest connected components of the adjacency matrices for motifs M_5 and M_8 have 127 and 128 nodes, respectively. Since both motif adjacency matrices contain the 62 nodes for which the ground truths are known, we consider only the sub-matrices corresponding to this set of 62 nodes when dealing with motifs M_5 and M_8 .

4.2 Performance Indexes

To assess the quality of the solutions, we use the following evaluation measures, well known in the literature:

NMI. The normalized mutual information $NMI(A, B)$ [5] of two divisions A and B of a network is defined as follows. Let C be the confusion matrix whose element C_{ij} is the number of nodes of community i of the partition A that are also in the community j of partition B .

$$NMI(A, B) = \frac{-2 \sum_{i=1}^{c_A} \sum_{j=1}^{c_B} C_{ij} \log(C_{ij}n / C_i.C_j)}{\sum_{i=1}^{c_A} C_i \log(C_i/n) + \sum_{j=1}^{c_B} C_j \log(C_j/n)} \quad (5)$$

where c_A (c_B) is the number of groups in partition A (B), C_i (C_j) is the sum of the elements of C in row i (column j), and n is the number of nodes. If $A = B$, $NMI(A, B) = 1$. If A and B are completely different, $NMI(A, B) = 0$.

F1 score. This measure [10] is calculated by using the Precision (P) and Recall (R) measures as $F1 = \frac{2RP}{R+P}$, where $P = \frac{TP}{TP+FP}$ and $R = \frac{TP}{TP+FN}$. True Positive (TP) refers to the number of nodes which are correctly assigned to communities,

Table 1. Florida Bay food web ground truths (GT1 and GT2) for the two large connected components of the motif *M6* adjacency matrix.

Node ID	Species	Component	GT1	GT2
8	'Benthic Phytoplankton'	1	Demersal Producer	Producer
9	'Thalassia'	1	Seagrass Producer	Producer
10	'Halodule'	1	Seagrass Producer	Producer
11	'Syringodium'	1	Seagrass Producer	Producer
13	'Drift Algae'	1	Algae Producer	Producer
14	'Epiphytes'	1	Algae Producer	Producer
24	'Benthic Flagellates'	1	Sediment Organism Microfauna	Microfauna
25	'Benthic Ciliates'	1	Sediment Organism Microfauna	Microfauna
26	'Meiofauna'	1	Sediment Organism Microfauna	Microfauna
29	'Other Cnidaridae'	1	Macroinvertebrates	Macroinvertebrates
30	'Echinoderma'	1	Macroinvertebrates	Macroinvertebrates
31	'Bivalves'	1	Macroinvertebrates	Macroinvertebrates
32	'Detritivorous Gastropods'	1	Macroinvertebrates	Macroinvertebrates
34	'Predatory Gastropods'	1	Macroinvertebrates	Macroinvertebrates
35	'Detritivorous Polychaetes'	1	Macroinvertebrates	Macroinvertebrates
36	'Predatory Polychaetes'	1	Macroinvertebrates	Macroinvertebrates
37	'Suspension Feeding Polych'	1	Macroinvertebrates	Macroinvertebrates
38	'Macrobenthos'	1	Macroinvertebrates	Macroinvertebrates
39	'Benthic Crustaceans'	1	Macroinvertebrates	Macroinvertebrates
40	'Detritivorous Amphipods'	1	Macroinvertebrates	Macroinvertebrates
41	'Herbivorous Amphipods'	1	Macroinvertebrates	Macroinvertebrates
42	'Isopods'	1	Macroinvertebrates	Macroinvertebrates
43	'Herbivorous Shrimp'	1	Macroinvertebrates	Macroinvertebrates
44	'Predatory Shrimp'	1	Macroinvertebrates	Macroinvertebrates
45	'Pink Shrimp'	1	Macroinvertebrates	Macroinvertebrates
48	'Detritivorous Crabs'	1	Macroinvertebrates	Macroinvertebrates
49	'Omnivorous Crabs'	1	Macroinvertebrates	Macroinvertebrates
50	'Predatory Crabs'	1	Macroinvertebrates	Macroinvertebrates
51	'Callinectes sapidus'	1	Macroinvertebrates	Macroinvertebrates
57	'Sardines'	1	Pelagic Fishes	Pelagic Fishes
58	'Anchovy'	1	Pelagic Fishes	Pelagic Fishes
59	'Bay Anchovy'	1	Pelagic Fishes	Pelagic Fishes
60	'Lizardfish'	1	Benthic Fishes	Benthic Fishes
61	'Catfish'	1	Benthic Fishes	Benthic Fishes
62	'Eels'	1	Demersal Fishes	Demersal Fishes
63	'Toadfish'	1	Benthic Fishes	Benthic Fishes
64	'Brotalus'	1	Demersal Fishes	Demersal Fishes
65	'Halfbeaks'	1	Pelagic Fishes	Pelagic Fishes
66	'Needlefish'	1	Pelagic Fishes	Pelagic Fishes
68	'Goldspotted killifish'	1	Demersal Fishes	Demersal Fishes
69	'Rainwater killifish'	1	Demersal Fishes	Demersal Fishes
72	'Silverside'	1	Pelagic Fishes	Pelagic Fishes
91	'Mullet'	1	Pelagic Fishes	Pelagic Fishes
93	'Blennies'	1	Benthic Fishes	Benthic Fishes
94	'Code Goby'	1	Benthic Fishes	Benthic Fishes
95	'Clown Goby'	1	Benthic Fishes	Benthic Fishes
96	'Flatfish'	1	Benthic Fishes	Benthic Fishes
99	'Other Pelagic Fishes'	1	Pelagic Fishes	Pelagic Fishes

(continued)

Table 1. (*continued*)

Node ID	Species	Component	GT1	GT2
100	'Omnivorous Ducks'	1	Demersal Fishes	Demersal Fishes
124	'Benthic POC'	1	Detritus	Detritus
15	'Free Bacteria'	2	Microbial Microfauna	Microfauna
16	'Water Flagellates'	2	Microbial Microfauna	Microfauna
17	'Water Cilitaes'	2	Microbial Microfauna	Microfauna
18	'Acartia Tonsa'	2	Zooplankton Microfauna	Microfauna
19	'Oithona nana'	2	Zooplankton Microfauna	Microfauna
20	'Paracalanus'	2	Zooplankton Microfauna	Microfauna
21	'Other Copepoda'	2	Zooplankton Microfauna	Microfauna
22	'Meroplankton'	2	Zooplankton Microfauna	Microfauna
23	'Other Zooplankton'	2	Zooplankton Microfauna	Microfauna
27	'Sponges'	2	Macroinvertebrates	Macroinvertebrates
123	'Water POC'	2	Detritus	Detritus
126	'Input'	2	Detritus	Detritus

False Positive (FP) refers to the nodes which are incorrectly assigned to communities, and False Negatives (FN) refers to the set of nodes which are incorrectly not assigned to the proper communities. F1 value reaches its best value at 1 and worst at 0.

Adjusted Rand Index. The Adjusted Rand Index (ARI) is a normalized version of the *Rand Index (RI)*[9] which simply assesses the degree of agreement between two partitions A and B . Let n_{11} be the number of pairs appearing in the same cluster in both A and B , n_{00} the number of pairs that appear in different clusters in both A and B , n_{10} the number of pairs appearing in the same cluster in A but in different clusters in B , and n_{01} the number of pairs that are in the same cluster in B and not in A . Then

$$ARI(A, B) = \frac{2(n_{00}n_{11} - n_{01}n_{10})}{(n_{00} + n_{01})(n_{01} + n_{11}) + (n_{00} + n_{10})(n_{10} + n_{11})} \tag{6}$$

4.3 Algorithms for Community Detection

We compare the two strategies of *M-MotifGA*, namely *MM-MotifGA*, in which the fitness function used is $\phi_{MM}(S)$, and *MS-MotifGA*, in which the fitness function is the sum of the single motif conductances, with the motifs-based *MRbi-part*, both in the case in which this last algorithm uses a single motif to detect communities and in the case of multiple motifs jointly used. We also compare *M-MotifGA* to two benchmark algorithms not using motifs: Louvain [4] and Infomap [15]. Louvain basically tries to optimize the modularity [12] of a partition through a greedy optimization technique. First, small communities are searched by optimizing modularity locally. Then, a new network whose nodes are the communities are built and these steps are repeated until a hierarchy of high-modularity communities is obtained. Infomap, on the contrary, exploits the principles of information theory characterizing the problem of community

detection as the problem of finding a description of minimum information of a random walk on the graph. Maximizing the Minimum Description Length [8] objective function, Infomap quickly provides an approximation of the optimal solution.

4.4 Results

Table 2(a)–(b) shows the *NMI*, *ARI* and *F1* values obtained for the two ground truths of the Florida Bay food web results. The statistical significance of the results has been checked by performing a t-test at the 5% significance level. The test rejected the null hypothesis that the values come from populations with equal means, and returned p-values, on average, below 0.1E-5.

For *MM-MotifGA* and *MS-MotifGA* we report both the average value and the standard deviation (in parenthesis) of the evaluation measures. For *MS-MotifGA*, we investigated as fitness function $\phi_{MS}(S) = \phi_{M_5}(S) + \phi_{M_6}(S) + \phi_{M_8}(S)$. On the ground truth *GT1*, *MM-MotifGA* outperforms all the other community detection schemes finding a number of communities ranging from 7 to 10. Similarly, *MS-MotifGA*, finds solutions with 7, 8, 9 and 10 communities, outperforming all the other methods. Considering *MS-MotifGA*, however, we observe that the strategy to sum the single motif conductances as fitness function to optimize, results in *NMI*, *ARI* and *F1* values lower than *MM-MotifGA*. As such, we conclude that if we explore separately the three motifs and then we recombine them by summing their conductances to obtain the function to optimize, the algorithm does not take into account the intersection which could exist between motifs in terms of edges. This intersection, as in the case of motifs M_5 and M_8 , and M_6 and M_8 for example, considered when jointly analyzing multiple motifs, is able to

Table 2. Florida Bay food web results.

		<i>MM-MotifGA</i>	<i>MS-MotifGA</i>	<i>MRbi-part</i> _{M_5}	<i>MRbi-part</i> _{M_6}
<i>GT1</i>	<i>NMI</i>	0.9241 (0.0756)	0.8602 (0.0781)	0.4392	0.504
	<i>ARI</i>	0.8451 (0.1923)	0.6879 (0.2141)	0.1388	0.3005
	<i>F1</i>	0.8765 (0.1489)	0.754 (0.1646)	0.3149	0.4437
<i>GT2</i>	<i>NMI</i>	0.8367 (0.1127)	0.6844 (0.1054)	0.3214	0.4822
	<i>ARI</i>	0.7039 (0.1798)	0.3886 (0.1106)	0.1045	0.3265
	<i>F1</i>	0.7756 (0.1329)	0.5549 (0.0727)	0.3087	0.4802

(a)

		<i>MRbi-part</i> _{M_8}	<i>MRbi-part</i> _{M_{MM}}	Louvain	Infomap
<i>GT1</i>	<i>NMI</i>	0.4197	0.3406	0.3879	0.4035
	<i>ARI</i>	0.1203	0.1291	0.2207	0.1423
	<i>F1</i>	0.2949	0.2962	0.4068	0.31
<i>GT2</i>	<i>NMI</i>	0.3573	0.2829	0.3034	0.3471
	<i>ARI</i>	0.1332	0.1241	0.2229	0.1592
	<i>F1</i>	0.3244	0.3101	0.434	0.3416

(b)

provide more meaningful communities, as the results show. Analyzing all the algorithms by Benson et al. where the number of communities has been set to 4, the communities found result in significantly lower values of the evaluation measures we considered. It is worth noting that considering only M_5 , M_6 or M_8 for clustering nodes does not produce satisfying results compared to our multiple-motifs strategies. Moreover, when jointly considering all the motifs as in $MRbi-part_{MM}$, the algorithm performs even worse than the single-motif strategies $MRbi-part_{M_5}$, $MRbi-part_{M_6}$, and $MRbi-part_{M_8}$. As such, we conclude that when the number of communities needs to be fixed in input as for $MRbi-part$, detecting clusters of multiple motifs using multiple-motifs conductance as the function to be minimized may lead to suboptimal results. Finally, comparing our method to *Louvain* and *Infomap*, which do not exploit motifs, we observe that also these methods are not able to find a good match with the ground truth. Focusing on the largest community of the ground truth (i.e., the *macroinvertebrates*) including 21 nodes, for example, we observe that *MM-MotifGA* perfectly matches it in all the runs of the algorithm. *Louvain* distributes the nodes into 4 different communities: 2 groups with 7 nodes are inserted into different communities, 3 nodes into another one, and the remaining nodes into another community. Finally, *Infomap* inserts all the 21 nodes into a unique but larger community including other nodes.

On the ground truth *GT2*, we obtain similar results. *MM-MotifGA* still outperforms all other methods, resulting in the highest *NMI*, *ARI* and *F1* values finding solutions with 5 or 6 communities. Overall, for all the algorithms, we observe *NMI*, *ARI* and *F1* values for *GT2* are lower than the values obtained for *GT1*. This behavior was also observed in our previous work [14] and it is probably due to the merging of some specie categories done on *GT2* to create macro-categories which do not perfectly reflect the modular structure of the network.

5 Conclusion

In this paper, we have proposed *M-MotifGA*, a method for discovering communities composed by multiple motifs. Based on a genetic algorithm, our method simultaneously considers different motifs for searching a partition with a number of communities minimizing the multiple-motifs conductance as fitness function. Simulations on the Florida bay food web network show that *M-MotifGA* results in *NMI*, *F1* and *ARI* values much higher than both the single-motif and the multiple-motif based analyzed strategies, *Louvain* and *Infomap*. Specifically, we have observed that for better matching the underlying real communities, not only multiple motifs should be simultaneously considered, but also fixing the number of communities to obtain as in Benson et al. [3] does not fully exploit the benefits of considering multiple motifs. As future work, we plan to extend our experiments to other datasets to further validate our method. We also intend to explore how community detection can be performed when several motifs appear at different network layers in multi-layered network structures.

References

1. Arenas, A., Fernández, A., Fortunato, S., Gómez, S.: Motif-based communities in complex network. *J. Phys. A: Math. Theor.* **42**(22), 224001 (2008)
2. Benson, A.R., Gleich, D.F., Leskovec, J.: Tensor spectral clustering for partitioning higher-order network structures. In: *Proceedings of the 2015 SIAM International Conference on Data Mining*, Vancouver, BC, Canada, 30 April–2 May 2015, pp. 118–126 (2015)
3. Benson, A.R., Gleich, D.F., Leskovec, J.: Higher-order organization of complex networks. *Science* **353**(6295), 163–166 (2016)
4. Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefevre, E.: Fast unfolding of communities in large networks. *J. Stat. Mech. Theory Exp.* **10**, P10008 (2008)
5. Cover, T.M., Thomas, J.A.: *Elements of Inf. Theory*, Wiley, Theory (1991)
6. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. In: *Proceedings of National Academy of Science. USA 1999*, pp. 7821–7826 (2002)
7. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston (1989)
8. Grünwald, P.D., Myung, I.J., Pitt, M.A.: *Advances in Minimum Description Length: Theory and Applications*. MIT Press, Cambridge (2005)
9. Hubert, L., Arabie, P.: Comparing partitions. *J. Classif.* **2**, 193–218 (1985)
10. Manning, C.D., Raghavan, P., Schütze, H., et al.: *Introduction to Information Retrieval*, vol. 1. Cambridge University Press, Cambridge (2008)
11. Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U.: Network motifs: simple building blocks of complex networks. *Science* **353**(298), 824–827 (2002)
12. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev.* **E69**, 026113 (2004)
13. Park, Y.J., Song, M.S.: A genetic algorithm for clustering problems. In: *Proceedings of 3rd Annual Conference on Genetic Algorithms*. Morgan Kaufmann Publishers, pp. 2–9 (1989)
14. Pizzuti, C., Socievole, A.: An evolutionary motifs-based algorithm for community detection. In *Proceedings of 8th IEEE International Conference on Information, Intelligences, Systems and Applications (IISA 2017)* (2017)
15. Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. *Proc. Natl. Acad. Sci.* **105**(4), 1118–1123 (2008)
16. Schaeffer, S.E.: Survey: graph clustering. *Comput. Sci. Rev.* **1**(1), 27–64 (2007)
17. Serrour, B., Arenas, A., Gómez, S.: Detecting communities of triangles in complex networks using spectral optimization. *Comput. Commun.* **34**(5), 629–634 (2011)
18. Ulanowicz, R.E., Bondavalli, C., Egnotovitch, M.S.: Network analysis of trophic dynamics in South Florida ecosystem, FY 97: the Florida Bay ecosystem. Annual Report to the United States Geological Service Biological Resources Division Ref. No.[UMCES] CBL, pp. 98–123 (1998)



Searching Relevant Variable Subsets in Complex Systems Using K-Means PSO

Gianluigi Silvestri¹, Laura Sani¹, Michele Amoretti¹, Riccardo Pecori^{1,2},
Emilio Vicari³, Monica Mordonini¹, and Stefano Cagnoni¹(✉)

¹ Dip. di Ingegneria e Architettura, Università di Parma, Parma, Italy
stefano.cagnoni@unipr.it

² SMARTTEST Research Centre, Università eCAMPUS, Novedrate, CO, Italy

³ CAMLIN Technologies Italy, Parma, Italy

Abstract. The Relevance Index method has been shown to be effective in identifying Relevant Sets in complex systems, i.e., variable sub-sets that exhibit a coordinated behavior, along with a clear independence from the remaining variables. The need for computing the Relevance Index for each possible variable sub-set makes such a computation unfeasible, as the size of the system increases. Because of this, smart search methods are needed to analyze large-size systems using such an approach. Niching metaheuristics provide an effective solution to this problem, as they join search capabilities to good exploration properties, which allow them to explore different regions of the search space in parallel and converge onto several local/global minima.

In this paper, we describe the application of a niching metaheuristic, K-means PSO, to a set of complex systems of different size, comparing, when possible, its results with the ground truth represented by the results of an exhaustive search, while we rely on the analysis of a domain expert to assess the results of larger systems. In all cases, we also compare the results of K-means PSO to another metaheuristic, based on a niching genetic algorithm, that we had previously developed.

Keywords: Complex systems · Relevant sets
Particle Swarm Optimization · K-means clustering

1 Introduction

Complex systems can be described by analyzing the collective behaviors and the emerging properties of their components, which are usually well-known and defined in terms of the system state variables. In several cases, however, the interactions among the elements of a system are not known in advance. Therefore, it is necessary to deduce some information about the organization of the system by observing the behavior of its relevant dynamic components.

In a previous work, Villani et al. proposed to identify candidate dynamical structures in complex systems [28], by means of a method, previously introduced

by Tononi et al. [26], for analyzing the coordinated behavior of sets of neurons in the brain cortex. Such a method detects subsets of the system variables that behave in a coordinated and coherent way, while loosely interacting with the remainder of the system. To do so, it associates to each of them an information theoretical measure, called *Relevance Index (RI)*. This measure can be normalized with respect to a reference system (termed *homogeneous system*), wherein the variables have the same marginal distribution as in the data set but are homogeneously correlated with each other. The normalized measure that can thus be computed, termed T_c index, quantifies how much a subset of variables of the system under investigation deviates from such a neutral condition.

The subsets can thus be ranked according to their T_c : the higher the T_c , the higher the correlation degree between the variables in a subset and the lower the interaction with the variables outside the subset. The most relevant sets, characterized by the highest T_c values, are referred to as *Candidate Relevant Sets (CRSs)*. In fact, the properly called *Relevant Subsets (RSs)* are CRSs that do not include (or are not included in) other CRSs with higher T_c values.

This means that a full description of a dynamical system requires that the T_c index be computed for each possible subset of the system variables. An exhaustive analysis becomes unfeasible as the dimension of the system increases, because the number of CRSs increases exponentially with the system size. The curse of “dimensionality” thus makes it impossible to analyze large systems exhaustively, even using massively parallel hardware such as GPUs, which fit the computational needs of this problem particularly well [27].

The contribution of this paper can be summarized with the following goals:

- optimizing algorithm efficiency by GPU-based parallel computations;
- studying custom versions of swarm intelligence algorithms to search for relevant sets as precisely and quickly as possible;
- comparing the aforementioned swarm intelligence algorithms with others previously applied to the same problems.

The rest of the paper comprises the background in Sect. 2, a description of the RI method in Sect. 3, and of the employed metaheuristics in Sect. 4, as well as a presentation of some interesting results in Sect. 5. Finally, Sect. 6 concludes the work.

2 Background

In general, unsupervised learning techniques aimed at inferring the emerging properties of a complex system are extremely attractive from the point of view of practical applications: for instance, they have been recently applied successfully to user profiling in the design of Intelligent Transportation Systems [7], or to the identification of the correct medical procedures based on medical dental records [4]. Previous works have already documented the use of information-theoretical measures as a possible solution to this clustering problem [12]. However, none of the existing methods has all the following desirable properties:

- the ability to identify groups of variables that change in a coordinated fashion;
- the ability to identify critical states;
- direct applicability to data, without any need to resort to models;
- robustness with respect to sampling effort and system size.

The Relevance Index (RI) appears to be a step forward towards a method having such features [29]. The RI is based on the Cluster Index (CI), introduced by Edelman et al. [26] and extends the applicability of the latter to a broad range of non-stationary dynamical systems, including abstract models of gene regulatory networks and simulated chemical [23,28], as well as biological [29] and social [11] systems.

In this work, we use Particle Swarm Optimization (PSO) as a search method, to counteract the complexity of RS search in complex system, which increases exponentially with the system size. PSO provides an effective solution to multi-faceted optimization problems and is often used for finding the optimal values for the parameters of other algorithms, such as hybrid SVM [17], deep belief neural networks [25], artificial bee colony [13], and the like.

The aim of standard PSO is to find a single point representing the global optimum of an N-dimensional function. However, in many multimodal function optimization problems, it is necessary to explore as many local minima as possible to improve the chances of finding the global optimum or even all the global optima, when more than one exist. Thus, to increase the particles' diversity degree, different techniques, which separately explore different regions of the search space, have been introduced in the literature. Among these, niching techniques assume a great importance. For example, they have been applied to a genetic algorithm for the estimation of the solar radiation [30], and to an evolutionary algorithm for forming collaborative learning teams in a class of students [32].

In the following we briefly summarize various use cases of modified and improved PSO algorithms, with a particular focus on niching PSO algorithms.

Introduced by Parsopoulos et al. [20], *objective function stretching* was one of the first strategies developed for analyzing multimodal functions. Its main purpose is to overcome the limitations of PSO due to untimely convergence to local solutions. The stretching approach modifies the fitness function to remove previously identified local optima. In this way, successive iterations of PSO can explore different regions of the research space and identify new solutions.

In 2002, Brits et al. introduced the *nbest PSO algorithm*, the first technique using parallel niching in particle swarm [6]. It is particularly suitable for finding multiple solutions in a system of equations. The same authors have also proposed another approach, which employs sub-swarms to locate multiple solutions in optimization problems of multi-modal functions, called *NichePSO* [5]. This algorithm uses a cognition-only model to evolve a main swarm that can generate sub-swarms each time a possible niche is identified.

Schoeman and Engelbrecht proposed a different niching approach [24], by implementing the *Vector-Based PSO*. This method identifies the niches by

exploiting some properties of the velocity vectors and some operations on them already present in PSO. Therefore, it does not require additional parameters.

Li proposed the *species-based PSO* [16], which encompasses the idea of classifying the population in groups of species. The definition of a species includes the seed, i.e., the particle with best fitness function, and the radius of the species, representing the Euclidean distance from the seed to the borders of the species. All particles inside this radius belong to the same species. A problem with this approach is the need to set the value of the radius, especially when the function presents niches of different dimensions.

Bird and Li proposed the *adaptive niching PSO* (ANPSO) [3], which removes the need to set the niche radius according to the specific problem and allows one to compute the parameters adaptively during execution. ANPSO uses an indirect graph to track the minimum distance between the particles, during the execution of the algorithm, and the niches are formed through sub-graphs that exclude all disconnected particles.

In this paper, we employ *K-means PSO* [21], where a clustering technique, namely *K-means* [18], is used to group the particles in sub-swarms. In these sub-swarms a local search is performed using the *gbest* topology, which uses the best individual's position as the only global attractor, and clustering is repeated at regular intervals.

K-means PSO, which is described in detail in Sect. 4, has been recently employed in different fields, such as clustering of satellite images [14], selecting effective features from the high-dimensional medical data set [10], etc. Moreover, K-means and PSO have been successfully hybridized in other contributions, not only in a sequential fashion, but also intertwined with each other [2] or in a dynamic and adaptive way [15].

3 Method

The RI can be used to study data from a wide range of dynamical system classes, with the purpose of identifying subsets of variables that behave in a somehow coordinated way, i.e., the variables belonging to the subset are integrated with each other much more than with the other variables not belonging to the subset. These subsets can be used to describe the whole system organization, thus they are named *relevant subsets*.

The computation of the RI, which is an information theoretical measure based on the well-known Shannon's entropy [9], is usually based on observational data, and probabilities are estimated as the relative frequencies of the values observed for each variable. The theoretical definition of the RI is summarized in the following.

The entropy $H(X)$ of a random variable X is defined as

$$H(X) = - \sum_x p(x) \log p(x) \quad (1)$$

The joint entropy of a pair of variables $H(X, Y)$ is defined as

$$H(X, Y) = - \sum_x \sum_y p(x, y) \log p(x, y) \quad (2)$$

Note that Eq. 2 can be naturally extended to sets of k elements.

Let us consider a system composed of n random variables X_1, X_2, \dots, X_n (e.g., agents, chemicals, genes, artificial entities, etc.) and suppose that S_k is a subset composed of k variables, with $k < n$. The RI of S_k is defined as:

$$RI(S_k) = \frac{I(S_k)}{MI(S_k; U \setminus S_k)}, \quad (3)$$

where I is the integration (multi-information), which measures the mutual dependence among the k elements in S_k , and MI is the mutual information, which measures the mutual dependence between subset S_k and the remaining part of the system $U \setminus S_k$.

The integration is defined as

$$I(S_k) = \sum_{s \in S_k} H(s) - H(S_k) \quad (4)$$

The mutual information $MI(S_k; U \setminus S_k)$ is defined as

$$MI(S_k; U \setminus S_k) = H(S_k) + H(U \setminus S_k) - H(S_k, U \setminus S_k) \quad (5)$$

The RI is undefined if $MI(S_k; U \setminus S_k) = 0$. However, a vanishing MI is a sign of separation of the subset under exam from the rest of the system, which suggests that the subset be studied separately.

We observe that the RI scales with the subset size, therefore a normalization method is required to compare RI values of subsets of different sizes. Moreover, the statistical significance of the differences of the RI should be assessed by means of an appropriate test. For these reasons, a statistical significance index was introduced [26]:

$$T_c(S_k) = \frac{RI(S_k) - \langle RI_h \rangle}{\sigma(RI_h)} = \frac{\nu RI - \nu \langle RI_h \rangle}{\nu \sigma(RI_h)} \quad (6)$$

where $\langle RI_h \rangle$ and $\sigma(RI_h)$ are, respectively, the average and the standard deviation of the RI of a sample of subsets of size k extracted from a reference homogeneous system U_h , and $\nu = \langle MI_h \rangle / \langle I_h \rangle$ is its normalization constant. A more detailed description of the method can be found in [23, 27].

4 Metaheuristic

In this paper, we use K-means PSO [21] for searching relevant dynamical structures of complex systems and for extracting the RSs when the dimension of

the variable space makes an exhaustive search unfeasible. K-means [18] is used as a niching technique to preserve diversity within the swarm, exploring many peaks in parallel. PSO, even if mainly utilized for continuous optimization, is employed in this case also on a discrete domain problem. The representation used in this paper (see Sect. 4.4) has already been successfully employed for feature selection [19].

The following subsections describe in detail the metaheuristic, the algorithms on which it is based, and their application to the analysis of complex systems.

4.1 Particle Swarm Optimization

Particle Swarm Optimization is a bio-inspired optimization algorithm based on the simulation of the social behavior of bird flocks. A swarm of s particles moves within a function domain (fitness function), searching for the optimum of the function (best fitness value).

Each particle of the swarm is characterized by:

- Position (x) in the search space;
- Fitness value at this position;
- Velocity (v), used to compute the next position;
- Memory (previous best) of the best position found so far by the particle;
- Fitness value of the previous best position.

The movement of each particle is influenced by its own best known position (previous best), but is also guided towards the best known positions of the other particles, which are updated as better values of the fitness function are found.

The particles thus move within the search space as a result of the following update steps:

$$\begin{cases} \vec{v}_i \leftarrow \chi(\vec{v}_i + \vec{U}(0, \phi_1) \otimes (\vec{p}_i - \vec{x}_i) + \vec{U}(0, \phi_2) \otimes (\vec{p}_g - \vec{x}_i)) \\ \vec{x}_i \leftarrow \vec{x}_i + \vec{v}_i \end{cases} \tag{7}$$

where χ is the so-called *constriction factor* [22], U are random variables uniformly distributed in $[0, \phi_i], i = 1, 2$, where ϕ_i 's (acceleration coefficients) are positive constants such that

$$\phi = \phi_1 + \phi_2 > 4 \tag{8}$$

$$\chi = \frac{2}{\phi - 2 + \sqrt{\phi^2 - 4\phi}} \tag{9}$$

Such constraints have been found to guarantee system stability (finite speed) [8].

The convergence of PSO on a Rastrigin function of two variables is shown in Fig. 1, which shows that the global optimum of the fitness function has been finally found and virtually the whole swarm has converged onto it. However, because of this, the local optima have not been identified. The enhancement of this process using a niching technique as K-means PSO allows the swarm to converge onto many peaks in parallel (see also Fig. 2 in Sect. 4.3).

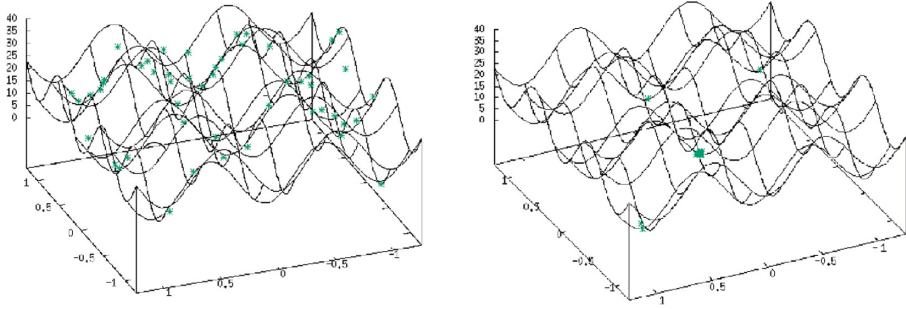


Fig. 1. Particle Swarm Optimization: the random initialization of the swarm positions, using a Rastrigin function of two variables as fitness function (left) and the results at convergence (right).

4.2 K-Means

K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster whose mean (centroid) is the nearest to it. K-means clustering uses the Euclidean distance as similarity criterion and

$$J = \sum_{k=1, N_c} \sum_{y^{(i)} \sim \omega_k} |y^{(i)} - \mu_k| \tag{10}$$

as the function to be optimized, where $y^{(i)}$ is the i^{th} sample, μ_k is the centroid of the k^{th} cluster, N_c is the number of clusters, and $y^{(i)} \sim \omega_k$ refers to all samples $y^{(i)}$ assigned to cluster k .

$M = \{\mu_1, \mu_2, \dots, \mu_{N_c}\}$ is the set of reference vectors, each of which represents the prototype for a class. J is minimized by computing μ_k as the sample mean of the data belonging to cluster k .

In practice, the algorithm partitions the input space S into k (the number of clusters, that must be set by the user) subspaces induced by the Euclidean distance. Each subspace s_i of S is defined as:

$$s_i = \{x_j \in S \mid d(x_j, \mu_i) = \min_t d(x_j, \mu_t)\} \tag{11}$$

This results in a partitioning of the data space into Voronoi cells (Voronoi tessellation).

4.3 K-Means PSO

With respect to the basic PSO algorithm, in the K-means PSO the search process is enhanced by a niching technique that maintains the diversity among the particles of the swarm and allows the swarm to explore and converge onto many peaks in parallel. In particular, in K-means PSO, at regular intervals, the K-means clustering algorithm [18] is applied to the swarm to reorganize it into

sub-swarms characterized by the proximity of their elements in the search space. The standard PSO algorithm is then independently applied to each sub-swarm thus identified.

The K-means PSO pseudo-code is reported in Algorithm 1. Lines 5, 6 and 7 represent the application of K-means clustering.

Algorithm 1. K-means PSO pseudo-code.

```

1: procedure  $\kappa$ PSO
2:   randomly initialize the particles' positions and velocities
3:   compute each particle's fitness
4:   for  $t = 1$  to  $T$  do                                     ▷  $T =$  number of iterations
5:     if  $t \bmod C = 0$  then                                   ▷ every  $C$  steps
6:       run K-means algorithm to identify niches
7:     end if
8:     update each particle's velocity                         ▷ as in standard PSO
9:     update each particle's position
10:    compute each particle's fitness
11:    update each particle's and each niche's best position
12:  end for
13: end procedure

```

With respect to standard PSO, K-means PSO requires some additional parameters: C is the number of PSO cycles to be performed between two clustering operations, k represents the number of clusters (it should be slightly higher than the number of local optima of the fitness function).

The convergence of K-means PSO using a Rastrigin function of two variables as fitness function is shown in Fig. 2 (right). The algorithm is able to explore many peaks in parallel, finding as many local optima as possible.

Figure 2 also compares the results of PSO and K-means PSO using a two-dimensional Rastrigin function as fitness function.

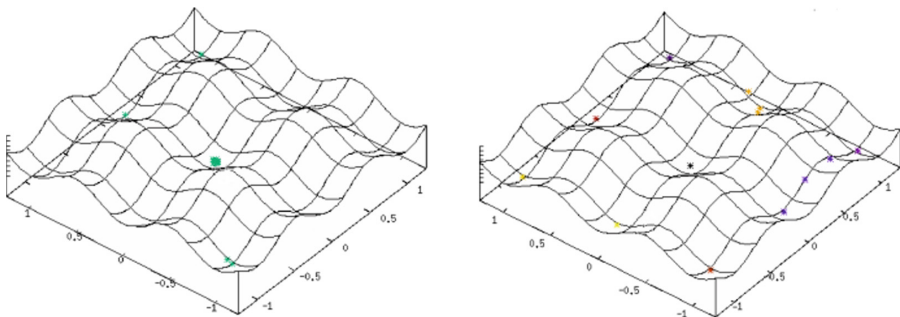


Fig. 2. Results of PSO (left) and K-means PSO (right) using a Rastrigin function of two variables as fitness function.

4.4 K-Means PSO for Searching Relevant Subsets

In our work, K-means PSO has been applied to the analysis of complex systems for detecting the highest- T_c CRSs in large-sized systems.

In our method, each particle i of the swarm represents CRSs as a binary string P_i of size N , where N is the number of variables that describe the system. A bit of this binary string is set to 1 if the corresponding variable is included in the CRS.

$$P_i(j) = \begin{cases} 1 & \text{if variable } j \text{ is included in the CRS} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

where $j \in [1, N]$.

Since PSO operates in \mathcal{R}^N and each particle i is therefore represented by its coordinates $p_r^i \in \mathcal{R}^N$, the corresponding binary vector P_i is obtained from each particle by setting the bits corresponding to positive coordinates in the search space to 1, and setting the others to 0.

$$P_i(j) = \begin{cases} 1 & \text{if } p_r^i(j) \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

The other steps of the algorithm, for example position and velocity update, are applied to the floating-point vectors p_r^i .

The fitness function to be maximized corresponds to the T_c index of the CRS associated to the particle and is implemented through a CUDA C [1] kernel that can compute in parallel the fitness values of large blocks of particles. Position and velocity updates have been parallelized as well.

A buffer has been introduced to store the best subsets (those having the highest T_c index) found during the run, and their corresponding fitness (T_c) values. Thus, at the end of the run, the best CRSs are not only the ones represented by the last swarm, but also the best ones found during the whole search process, which are stored in the buffer.

5 Experimental Results

The K-means PSO has been evaluated on a set of meaningful systems described by Boolean variables. The results have been compared with those achieved by an exhaustive search, when computationally feasible, and by another hybrid meta-heuristic, based on genetic algorithms and local search, which we had previously developed [23]. All the algorithms rely on the same GPU implementation of the fitness function.

Given the stochastic nature of the two meta-heuristics, 30 independent runs of the algorithm were executed to assess its performance.

The results are summarized in Table 1. The first case study is a simulation of a chemical system called Catalytic Reaction System (CatRS), featuring 26 variables. The second one is a stochastic artificial system reproducing a Leaders & Followers (LF) behavior, described by 28 variables. In the third example,

denoted as Green Community Network (GCN), the data come from a real world environment (participation of partners in project meetings) and are described by 56 variables, a size for which an exhaustive search is unfeasible on a standard computer, even using GPU parallelization. Therefore, it was analyzed only by the two meta-heuristics.

Table 1. Results obtained by K-means PSO (K) on three different systems and comparison with [23] in terms of time and speed-up with respect to an exhaustive search (E).

System	Size	N. data	Time[s] (E)	Time[s] [23]	Time[s] (K)	Speed-up [23]	Speed-up (K)
CatRS	26	751	53	24	6	2.2	8.8
LF	28	150	196	19	2	10.3	98
GCN	56	124	n.a.	71	18	n.a.	n.a.

Tests were run on a Linux server equipped with a 1.6 GHz Intel I7 CPU, 6 GB of RAM and a GeForce GTX 680 GPU by NVIDIA.

The parameters regulating the behavior of K-means PSO have been set as reported in Table 2. They are the ones that led to the best results. Moreover, ϕ_1 , ϕ_2 and χ were obtained considering also the constraints presented in Eqs. 8 and 9.

Table 2. K-means PSO parameter settings. The parameters are defined in Sect. 4.

System	s	T	k	C	ϕ_1	ϕ_2	χ
CatRS	2000	501	10	20	2.05	2.05	0.73
LF	1000	501	10	20	2.05	2.05	0.73
GCN	2000	2001	10	20	2.05	2.05	0.73

The results of the two metaheuristics have been evaluated both in terms of quality and of speed-up with respect to an exhaustive search. Quality has been evaluated, when feasible, counting the number of highest- T_c CRSs detected by the exhaustive search, but not by K-means PSO. Regarding the large-sized system, for which the results of the exhaustive search were not available, we have relied on the opinion of an expert to assess their quality.

The results obtained with the smaller-size systems, for which the comparison is possible, are almost always the same as those provided by an exhaustive search. Only at most one out of the top 50 sets was not detected by both metaheuristics, and we considered this acceptable and more than enough to understand the main dynamics of the systems. For this reason we were able to compare the results of the two metaheuristics with those of the exhaustive search.

The results obtained on the GCN were judged as reasonable by an expert. In this case, too, the same results were obtained in different runs, with marginal occasional differences only within the least significant sets.

K-means PSO exhibits both good exploration capabilities, thanks to its niching behavior, and fast convergence. The latter is probably justified by the nature of the problem that is such that *dominant* variables exist, which are repeatedly present in the relevant sets. In other words, K-means PSO, which converges extremely fast when solving optimization problems with separable functions, but struggles when dealing with strongly-dependent variables, can easily find those variables that are dominant across most groups and rapidly converges onto the most relevant groups that include them.

This makes it possible for K-means PSO to achieve a significant speed-up with respect to both the exhaustive search and the hybrid meta-heuristic. Basically, the lower speed-up of the latter is partially due to the presence of the local search, which ensures a better exploration of the neighborhoods of the local minima. In the tests we have made, however, K-means PSO has not suffered from the lack of such a feature.

It is worth highlighting again that all methods rely on the same GPU implementation of the fitness function, which means that the observed differences depend only on the efficiency and complexity of the algorithms and not on their implementation.

6 Conclusion and Future Developments

Finding hidden relationships between variables in complex systems is a very relevant, but computationally heavy, task. Evolutionary and swarm intelligence algorithms are very good candidate solutions for extracting such relationships using the Relevance Index method, when the size of the system under consideration is large.

The work described in this paper is mainly related with the study of custom versions of swarm intelligence algorithms to search for relevant sets as precisely and quickly as possible. In particular we employed K-means PSO for searching relevant sets in a number of complex systems, comparing the results obtained with such an algorithm with those obtained by HyReSS, a genetic algorithm-based metaheuristic we had previously developed [23].

K-means PSO achieves a very good compromise between efficiency and precision in detecting the relevant sets. Compared to HyReSS, it appears to be generally quicker but slightly less precise, because of the presence, in the latter, of a local search step.

As future work we plan to study possible extensions of the method to other application fields. In particular, we are considering applications, for example in pattern recognition, in which we foresee that quickly finding most, even if not all, the relevant sets could be enough to obtain significant results.

Acknowledgments. The work of Michele Amoretti was supported by the University of Parma Research Fund - FIL 2016 - Project “NEXTALGO: Efficient Algorithms for Next-Generation Distributed Systems”.

The authors would like to thank Andrea Roli, Roberto Serra, and Marco Villani for the enlightening discussions and comments on this work.

References

1. CUDA Toolkit. <http://developer.nvidia.com/cuda-toolkit>. Accessed 12 Mar 2018
2. Atabay, H.A., Sheikhzadeh, M.J., Torshizi, M.: A clustering algorithm based on integration of K-means and PSO. In: 2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), pp. 59–63, March 2016
3. Bird, S., Li, X.: Adaptively choosing niching parameters in a PSO. In: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, GECCO 2006, pp. 3–10. ACM, New York (2006)
4. Bokhari, S.M.A., Basharat, I., Khan, S.A., Qureshi, A.W., Ahmed, B.: A framework for clustering dental patients’ records using unsupervised learning techniques. In: 2015 Science and Information Conference (SAI), pp. 386–394, July 2015
5. Brits, R., Engelbrecht, A., van den Bergh, F.: A niching particle swarm optimizer. In: 4th Asia-Pacific Conference on Simulated Evolution and Learning, pp. 692–696, January 2002
6. Brits, R., Engelbrecht, A.P., van den Bergh, F.: Solving systems of unconstrained equations using particle swarm optimization. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 3, p. 6, October 2002
7. Canale, S., Giorgio, A.D., Lisi, F., Panfilì, M., Celsi, L.R., Suraci, V., Priscoli, F.D.: A future internet oriented user centric extended intelligent transportation system. In: 2016 24th Mediterranean Conference on Control and Automation (MED), pp. 1133–1139, June 2016
8. Clerc, M., Kennedy, J.: The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.* **6**(1), 58–73 (2002)
9. Cover, T., Thomas, A.: *Elements of Information Theory*, 2nd edn. Wiley-Interscience, New York (2006)
10. Doreswamy, Salma, M.U.: PSO based fast K-means algorithm for feature selection from high dimensional medical data set. In: 2016 10th International Conference on Intelligent Systems and Control (ISCO), pp. 1–6, January 2016
11. Filisetti, A., Villani, M., Roli, A., Fiorucci, M., Poli, I., Serra, R.: On some properties of information theoretical measures for the study of complex systems. In: Pizutti, C., Spezzano, G. (eds.) WIVACE 2014. CCIS, vol. 445, pp. 140–150. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12745-3_12
12. Gershenson, C., Fernandez, N.: Complexity and information: measuring emergence, self-organization, and homeostasis at multiple scales. *Complexity* **18**(2), 29–44 (2012)
13. Goudarzi, S., Hassan, W.H., Anisi, M.H., Soleymani, A., Sookhak, M., Khan, M.K., Hashim, A.H.A., Zareei, M.: ABC-PSO for vertical handover in heterogeneous wireless networks. *Neurocomputing* **256**(Supplement C), 63–81 (2017). *Fuzzy Neuro Theory and Technologies for Cloud Computing*
14. Kumar, G., Sarth, P.P., Ranjan, P., Kumar, S.: Satellite image clustering and optimization using K-means and PSO. In: 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), pp. 1–4, July 2016

15. Li, H., He, H., Wen, Y.: Dynamic particle swarm optimization and K-means clustering algorithm for image segmentation. *Optik-Int. J. Light Electron Opt.* **126**(24), 4817–4822 (2015)
16. Li, X.: Adaptively choosing neighbourhood bests using species in a particle swarm optimizer for multimodal function optimization. In: Deb, K. (ed.) *GECCO 2004*. LNCS, vol. 3102, pp. 105–116. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24854-5_10
17. Liu, B., Li, Z.: Study on the automatic recognition of hidden defects based on Hilbert Huang transform and hybrid SVM-PSO model. In: *2017 Prognostics and System Health Management Conference (PHM-Harbin)*, pp. 1–7, July 2017
18. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297 (1967)
19. Nguyen, H.B., Xue, B., Andreae, P.: Mutual information for feature selection: estimation or counting? *Evol. Intel.* **9**(3), 95–110 (2016)
20. Parsopoulos, K.E., Plagianakos, V.P., Magoulas, G.D., Vrahatis, M.N.: Improving the particle swarm optimizer by function “stretching”. In: Hadjisavvas, N., Pardalos, P.M. (eds.) *Advances in Convex Analysis and Global Optimization. Nonconvex Optimization and Its Applications*, pp. 445–457. Springer, Boston (2001). https://doi.org/10.1007/978-1-4613-0279-7_28
21. Passaro, A., Starita, A.: Particle swarm optimization for multimodal functions: a clustering approach. *J. Artif. Evol. Appl.* **2008**, 15 p. (2008). <https://doi.org/10.1155/2008/482032>. Article ID 482032
22. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. *Swarm Intell.* **1**(1), 33–57 (2007)
23. Sani, L., et al.: Efficient search of relevant structures in complex systems. In: Adorni, G., Cagnoni, S., Gori, M., Maratea, M. (eds.) *AI*IA 2016*. LNCS (LNAI), vol. 10037, pp. 35–48. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49130-1_4
24. Schoeman, I.L.: Niching in particle swarm optimization. Ph.D. thesis, School of Engineering, University of Pretoria (2010)
25. Sun, Q., Wang, Y., Jiang, Y., Shao, L., Chen, D.: Fault diagnosis of SEPIC converters based on PSO-DBN and wavelet packet energy spectrum. In: *2017 Prognostics and System Health Management Conference (PHM-Harbin)*, pp. 1–7, July 2017
26. Tononi, G., McIntosh, A., Russel, D., Edelman, G.: Functional clustering: identifying strongly interactive brain regions in neuroimaging data. *Neuroimage* **7**, 133–149 (1998)
27. Vicari, E., et al.: GPU-based parallel search of relevant variable sets in complex systems. In: Rossi, F., Piotto, S., Concilio, S. (eds.) *WIVACE 2016*. CCIS, vol. 708, pp. 14–25. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57711-1_2
28. Villani, M., Filisetti, A., Benedettini, S., Roli, A., Lane, D., Serra, R.: The detection of intermediate level emergent structures and patterns. In: Liò, P., Miglino, O., Nicosia, G., Nolfi, S., Pavone, M. (eds.) *Proceedings of ECAL 2013, the 12th European Conference on Artificial Life*. MIT Press (2013)
29. Villani, M., Roli, A., Filisetti, A., Fiorucci, M., Poli, I., Serra, R.: The search for candidate relevant subsets of variables in complex systems. *Artif. Life* **21**(4), 412–431 (2015)
30. Will, A., Bustos, J., Bocco, M., Gotay, J., Lamelas, C.: On the use of niching genetic algorithms for variable selection in solar radiation estimation. *Renew. Energy* **50**, 168–176 (2013)

31. Xue, B., Zhang, M., Browne, W.N., Yao, X.: A survey on evolutionary computation approaches to feature selection. *IEEE Trans. Evol. Comput.* **20**(4), 606–626 (2016)
32. Yannibelli, V., Amandi, A.: A deterministic crowding evolutionary algorithm to form learning teams in a collaborative learning context. *Expert Syst. Appl.* **39**(10), 8584–8592 (2012)

Author Index

- Abou-Hassan, Ali 16
Alberghina, Lilia 165
Amoretti, Michele 129, 308
Anzoise, Valentina 229
- Bahrudeen, Mohamed N. M. 181
Baiolletti, Marco 271
Bellotti, Roberto 257
Birattari, Mauro 243
Braccini, Michele 104, 116
Budroni, Marcello A. 32
Busti, Stefano 165
- Cagnoni, Stefano 129, 308
Cardellicchio, Angelo 257
Carletti, Timoteo 3
Castiglione, Filippo 88
Concilio, Simona 49
- D'Ambrosio, Raffaele 59
Damiano, Luisa 73
Della Marra, Fabio 197
Di Biasi, Luigi 49
- Guaragnella, Cataldo 257
- Khoroshiltseva, Marina 284
- Ligot, Antoine 243
Lombardi, Angela 257
- Mameli, Valentina 284
Marchettini, Nadia 32
Milani, Alfredo 271
Moccaldi, Martina 59
Montagna, Sara 104
Mordonini, Monica 129, 308
Musa, Martina 153
- Nicolay, Delphine 3
- Oliveira, Samuel M. D. 181
- Palumbo, Maria Concetta 88
Palumbo, Pasquale 165
Papa, Federico 165
Paternoster, Beatrice 59
Pecori, Riccardo 129, 308
Pedicini, Marco 88
Piotto, Stefano 49
Pizzuti, Clara 296
Poli, Irene 229, 284
- Ribeiro, Andre S. 181
Righi, Riccardo 212
Ristori, Sandra 16
Roli, Andrea 104, 116, 243
Rossi, Federico 16, 32, 59
Rustici, Mauro 32
- Samoili, Sofia 212
Sani, Laura 129, 308
Santucci, Valentino 271
Sapienza, Davide 142
Serra, Roberto 116, 129, 142, 153
Sessa, Lucia 49
Silvestri, Gianluigi 308
Slanzi, Debora 229, 284
Socievole, Annalisa 296
Stano, Pasquale 73
Startceva, Sofia 181
- Tangaro, Sabina 257
Teusink, Bas 165
Torbensen, Kristian 16
- Vanoni, Marco 165
Vicari, Emilio 129, 308
Villani, Marco 116, 129, 142, 153
- Wortel, Meike 165