

Ningchuan Xiao
Mei-Po Kwan
Michael F. Goodchild
Shashi Shekhar (Eds.)

LNCS 7478

Geographic Information Science

7th International Conference, GIScience 2012
Columbus, OH, USA, September 2012
Proceedings

 Springer

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

TU Dortmund University, Germany

Madhu Sudan

Microsoft Research, Cambridge, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max Planck Institute for Informatics, Saarbruecken, Germany

Ningchuan Xiao Mei-Po Kwan
Michael F. Goodchild Shashi Shekhar (Eds.)

Geographic Information Science

7th International Conference, GIScience 2012
Columbus, OH, USA, September 18-21, 2012
Proceedings



Springer

Volume Editors

Ningchuan Xiao

Ohio State University, Department of Geography
1036 Derby Hall, 154 N Oval Mall, Columbus, OH 43210, USA
E-mail: xiao.37@osu.edu

Mei-Po Kwan

University of Illinois at Urbana-Champaign
Department of Geography and Geographic Information Science
Urbana, IL 61801, USA
and

Hong Kong Polytechnic University
Department of Land Surveying and Geo-Informatics
Hung Hom, Kowloon, Hong Kong
E-mail: mpk654@gmail.com

Michael F. Goodchild

University of California, Department of Geography
Santa Barbara, CA 93106-4060, USA
E-mail: good@geog.ucsb.edu

Shashi Shekhar

University of Minnesota, Department of Computer Science
Minneapolis, MN 55455, USA
E-mail: shekhar@cs.umn.edu

ISSN 0302-9743

e-ISSN 1611-3349

ISBN 978-3-642-33023-0

e-ISBN 978-3-642-33024-7

DOI 10.1007/978-3-642-33024-7

Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012945032

CR Subject Classification (1998): H.2.8, H.4, I.2.1, I.2.4, F.2.2, H.2.7

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

© Springer-Verlag Berlin Heidelberg 2012

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, 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.

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

In the past decade, the GIScience conference series has become home to researchers from fields such as cognitive science, computer science, engineering, geography, information science, mathematics, philosophy, psychology, social science, environmental sciences, and statistics. Starting in 2000, the conference series has attracted a highly interdisciplinary community of GIScience researchers from academia, industry, and government, who have brought exciting basic research findings to the meetings and advanced our knowledge in all aspects of geographic information science. Since 2006, the conference series has adopted a transatlantic pattern. Following the highly successful Zurich meeting, the conference was hosted this year at Columbus, Ohio, the heart of the American Midwest where many great things have taken place since the mid-1950s.

GIScience 2012 followed the conference tradition of a two-track pattern consisting of full papers of up to 14 pages and extended abstracts of 1,500 words that describe work in progress. This pattern has proven to be rather flexible in accommodating different cultures in the GIScience community. This year we received 57 full-paper submissions and each paper was reviewed by at least three Program Committee members. Among these submissions, 26 were accepted and included in this volume of *Lecture Notes in Computer Science*. These papers were included in the conference program for oral presentation. We also received 127 extended abstract submissions for a later deadline. After being reviewed by at least two committee members, 77 extended abstracts were accepted for oral presentation and 28 for poster presentation.

The accepted full papers and extended abstracts represent a broad range of cutting-edge research in GIScience. While the traditional research topics are well reflected in these papers, emerging topics that involve new research hot-spots such as cyberinfrastructure, big data, and Web-based computing also occupied a significant portion of the conference program. In addition to the papers, we had six keynote speakers and a number of workshops and tutorials. We noted the increasing participation in GIScience from Asian countries such as China, India, Japan, and Korea, in both the Program Committee and paper submissions. We are confident that the conference participants experienced an exciting program in Columbus.

None of the above-mentioned events would have happened without help from the Program Committee. We would like to thank all the committee members for their time and hard work in making it all possible. We are grateful to Oscar Larson at the Association of American Geographers for efficiently organizing the conference events; we would have been lost without his insight in conference management. We appreciate the help from the Zurich team, Sara Fabrikant, Ross Purves,

and Robert Weibel, who organized the 2010 GIScience conference. We thank Tom Cova for his constant help during the preparation process. We also thank all the sponsors who supported various events and activities of the conference. Mostly important, we would like to thank our participants for submitting their important work and making exciting contributions to the conference.

July 2012

Ningchuan Xiao
Mei-Po Kwan
Michael Goodchild
Shashi Shekhar

Organization

General Chair

Mei-Po Kwan
University of Illinois at Urbana-Champaign and
Hong Kong Polytechnic University

Program Chair

Ningchuan Xiao
Ohio State University

Program Co-chairs (Full Papers)

Michael Goodchild
Shashi Shekhar
University of California, Santa Barbara
University of Minnesota

Program Co-chair (Extended Abstracts)

Hui Lin
Chinese University of Hong Kong

Workshop Chair

Ola Ahlqvist
Ohio State University

Program Committee

Ola Ahlqvist	Ohio State University
Luc Anselin	Arizona State University
Marc Armstrong	University of Iowa
Kate Beard-Tisdale	University of Maine
Itzhak Benenson	Tel Aviv University
David Bennett	University of Iowa
Michela Bertolotto	University College Dublin
Ling Bian	University at Buffalo
Thomas Bittner	University at Buffalo
Dan Brown	University of Michigan
Barbara Battenfield	University of Colorado
Gilberto Camara	INPE, Brazil
Nicholas Chrisman	Laval University

Christophe Claramunt	NARI, France
Keith Clarke	UC Santa Barbara
Helen Couclelis	UC Santa Barbara
Tom Cova	University of Utah
Martin Dodge	University of Manchester
Juergen Doellner	Hasso Plattner Institut, Potsdam
Matt Duckham	University of Melbourne
Jason Dykes	City University London
Max Egenhofer	University of Maine
Sara Irina Fabrikant	University of Zurich
Peter Fisher	University of Leicester
Mark Gahegan	University of Auckland
Rina Ghose	University of Wisconsin-Milwaukee
Peng Gong	University of California, Berkeley
Michael Goodchild	University of California, Santa Barbara
Ian Gregory	University of Lancaster
Dan Griffith	University of Texas, Dallas
Diansheng Guo	University of South Carolina
Muki Haklay	University College London
Lars Harrie	Lund University
Francis Harvey	University of Minnesota
Gerard Heuvelink	Wageningen University
Stephen Hirtle	University of Pittsburgh
Piotr Jankowski	San Diego State University
Bin Jiang	University of Gävle, Sweden
Christopher Jones	Cardiff University
Derek Karssenber	Utrecht University
Maggi Kelly	University of California, Berkeley
Alexander Klippel	Pennsylvania State University
Menno-Jan Kraak	ITC, The Netherlands
Werner Kuhn	University of Münster
Mei-Po Kwan	University of Illinois at Urbana-Champaign and Hong Kong Polytechnic University
Phaedon Kyriakidis	UC Santa Barbara
Nina Lam	Louisiana State University
Jiyeong Lee	University of Seoul
Brian Lees	Australian Defence Force Academy
Ron Li	Ohio State University
Xia Li	Sun Yat-sen University
Xiang Li	East China Normal University
Hui Lin	Chinese University of Hong Kong
Lin Liu	University of Cincinnati
Yu Liu	Peking University
Amy Lobben	University of Oregon

Paul Longley	University College London
Feng Lu	Chinese Academy of Sciences
Xiujun Ma	Peking University
Jeremy Mennis	Temple University
Carolyn Merry	Ohio State University
Harvey Miller	University of Utah
Daniel R. Montello	UC Santa Barbara
Alan Murray	Arizona State University
Tomoki Nakaya	Ritsumeikan University, Japan
Nora Newcombe	Temple University
Fangqu Niu	Chinese Academy of Sciences
David O'Sullivan	University of Auckland
Atsuyuki Okabe	University of Tokyo
Antonio Paez	McMaster University
Dimitris Papadias	UST, Hong Kong
Karin Pfeffer	University of Amsterdam
Lilian Pun	Hong Kong Polytechnic University
Ross Purves	University of Zurich
Martin Raubal	ETH Zurich
Maria Andrea Rodríguez-Tastets	Universidad de Concepción
Anne Ruas	Institut Géographie National
Nadine Schuurman	Simon Fraser University
Raja Sengupta	IIT-Delhi
Monika Sester	Leibniz University Hannover
Shih-Lung Shaw	University of Tennessee
Shashi Shekhar	University of Minnesota
Wenzhong Shi	Hong Kong Polytechnic
Takeshi Shirabe	Royal Institute of Technology, Sweden
Ashton Shortridge	Michigan State University
Renee Sieber	McGill University
Jack Snoeyink	University of North Carolina
Emmanuel Stefanakis	University of New Brunswick
Kathleen Stewart	University of Iowa
Jean-Claude Thill	University of North Carolina at Charlotte
Paul Torrens	University of Maryland
Ming-Hsiang Tsou	San Diego State University
Marc van Kreveld	Utrecht University
Monica Wachowicz	University of New Brunswick
Shaowen Wang	University of Illinois
Robert Weibel	University of Zurich
John Wilson	University of Southern California
Stephan Winter	University of Melbourne
Michael Worboys	University of Maine
Dawn Wright	ESRI and Oregon State University

Ningchuan Xiao	Ohio State University
Yichun Xie	Eastern Michigan University
Chaowei Yang	George Mason University
Keiji Yano	Ritsumeikan University
Bailang Yu	East China Normal University
May Yuan	University of Oklahoma
Chenghu Zhou	Chinese Academy of Sciences

Additional Reviewers

Jared Aldstadt	University at Buffalo
Desheng Liu	Ohio State University
Wenwu Tang	University of North Carolina, Charlotte

Sponsors

Environment System Research Institute (ESRI)
Department of Land Surveying and Geo-Informatics, Hong Kong Polytechnic
University
Department of Geography, Ohio State University
Association of American Geographers (AAG)
State key Laboratory of Resource and Environment Information System,
Institute of Geographical Sciences and Natural Resource Research,
Chinese Academy of Sciences
Ohio Supercomputer Center
Annals of Association of American Geographers
Department of Civil and Environmental Engineering and Geodetic Science
Center for Mapping, Ohio State University
International Journal of Geographical Information Science
Institute for Population Research, Ohio State University
University Consortium of Geographic Information Science

Table of Contents

Combining Trip and Task Planning: How to Get from A to <i>Passport</i> . . .	1
<i>Amin Abdalla and Andrew U. Frank</i>	
Automated Centerline Delineation to Enrich the National Hydrography Dataset	15
<i>Chris Anderson-Tarver, Mike Gleason, Barbara Buttenfield, and Larry Stanislawski</i>	
Evolution Strategies for Optimizing Rectangular Cartograms	29
<i>Kevin Buchin, Bettina Speckmann, and Sander Verdonschot</i>	
Context-Aware Similarity of Trajectories	43
<i>Maïke Buchin, Somayeh Dodge, and Bettina Speckmann</i>	
Generating Named Road Vector Data from Raster Maps	57
<i>Yao-Yi Chiang and Craig A. Knoblock</i>	
An Ordering of Convex Topological Relations	72
<i>Matthew P. Dube and Max J. Egenhofer</i>	
Toward Web Mapping with Vector Data	87
<i>Julien Gaffuri</i>	
spatial@linkedsience – Exploring the Research Field of GIScience with Linked Data	102
<i>Carsten Kestler, Krzysztof Janowicz, and Tomi Kauppinen</i>	
Crowdsourcing Satellite Imagery Analysis: Study of Parallel and Iterative Models	116
<i>Nicolas Maisonneuve and Bastien Chopard</i>	
Quantifying Resolution Sensitivity of Spatial Autocorrelation: A Resolution Correlogram Approach	132
<i>Pradeep Mohan, Xun Zhou, and Shashi Shekhar</i>	
<i>LocalAlert</i> : Simulating Decentralized Ad-Hoc Collaboration in Emergency Situations	146
<i>Silvia Nittel, Christopher Dorr, and John C. Whittier</i>	

High-Level Event Detection in Spatially Distributed Time Series	160
<i>Avinash Rude and Kate Beard</i>	
Towards Vague Geographic Data Warehouses	173
<i>Thiago Luís Lopes Siqueira, Cristina Dutra de Aguiar Ciferri, Valéria Cesário Times, and Ricardo Rodrigues Ciferri</i>	
Measuring the Influence of Built Environment on Walking Behavior: An Accessibility Approach	187
<i>Guibo Sun, Hui Lin, and Rongrong Li</i>	
Social Welfare to Assess the Global Legibility of a Generalized Map	198
<i>Guillaume Touya</i>	
Investigations into the Cognitive Conceptualization and Similarity Assessment of Spatial Scenes	212
<i>Jan Oliver Wallgrün, Jinlong Yang, Alexander Klippel, and Frank Dylla</i>	
A Qualitative Bigraph Model for Indoor Space	226
<i>Lisa A. Walton and Michael Worboys</i>	
Dynamic Refuse Collection Strategy Based on Adjacency Relationship between Euler Cycles	241
<i>Toyohide Watanabe and Kosuke Yamamoto</i>	
Impact of Indoor Location Information Reliability on Users' Trust of an Indoor Positioning System	258
<i>Ting Wei and Scott Bell</i>	
Ontology for the Engineering of Geospatial Systems	270
<i>Nancy Wiegand</i>	
Preserving Detail in a Combined Land Use Ontology	284
<i>Nancy Wiegand</i>	
The Maptree: A Fine-Grained Formal Representation of Space	298
<i>Michael Worboys</i>	
Automatic Creation of Crosswalk for Geospatial Metadata Standard Interoperability	311
<i>Hui Yang and Gefei Feng</i>	
A Dartboard Network Cut Based Approach to Evacuation Route Planning: A Summary of Results	325
<i>KwangSoo Yang, Venkata M.V. Gunturi, and Shashi Shekhar</i>	

Hybrid Geo-spatial Query Methods on the Semantic Web with a Spatially-Enhanced Index of DBpedia	340
<i>Eman M.G. Younis, Christopher B. Jones, Vlad Tanasescu, and Alia I. Abdelmoty</i>	
Extracting Dynamic Urban Mobility Patterns from Mobile Phone Data	354
<i>Yihong Yuan and Martin Raubal</i>	
Author Index	369

Combining Trip and Task Planning: How to Get from A to *Passport*

Amin Abdalla and Andrew U. Frank

Vienna University of Technology
Institute for Geoinformation and Cartography
{abdalla, frank}@geoinfo.tuwien.ac.at

Abstract. Navigation-tools currently give us directions from location A to B. They help us with the physical process of moving from here to there. Tasks in general, are achieved by the subsequent determination and execution of sub-tasks until the goal is achieved. To help achieve the higher-ranking task, we commonly use so called “personal information management”-tools (PIM-tools). They offer possibilities to manage and organize information about errands that have personal or social implications. Such tasks are described in informal ways, todo-lists for example offer the storage of textual description of an errand, sometimes allowing geographic or temporal information to be added. The paper proposes a formalism that can produce instructions leading from A to the fulfilment of the “task”. Thus connecting the high-level task, that represents intentions, with the physical level of navigation.

Keywords: PIM, task planning, routing, LBS, shortest path.

1 Introduction

The problem of how humans find their way from A to B is an issue that puzzled researchers for many years and is investigated in various domains [24, 26]. As a consequence we now see tools helping us to plan and execute navigation from one location to another. But these tools know little about the purpose of our trips. Although human wayfinding can, as Golledge and Gärling put it, be regarded as an “...*purposive, directed and motivated activity...*” [11] we have not seen attempts to integrate our intentions into GIS or navigation tools. The motivation behind our trips is an important factor that can help to improve the usability of such tools. Imagine asking your navigation device what the next task is and how to achieve it.

To handle and manage information about tasks or errands we usually use calendars or todo-lists, increasingly in digital form. These tools essentially store information about our intentions and motivations that imply sub-tasks such as navigation. The research field of managing and organizing personal information is referred to as “personal information management” (PIM) [16] or “task information management” [19]. While most of the studies are concerned about how to maintain digital information such as documents, pictures or webpages,

efficiently; we like to focus on the task management part of it. That is how we retain information about our intentions and future activities and more specific, on how we plan to execute them.

In the following pages we will investigate the issue of integrating tasks into GIS respectively navigation-applications (or vice versa). By examining a specific problem we try to determine what information is needed to compute a suitable path through the search space (i.e.: the set of all possible states) of the problem.

Starting with an overview of relevant work, we will present and analyze the example of a “passport application”. Finally we will propose a solution by narrowing it down to a shortest path problem, such that we can give instructions of how to get from a specific location and situation to the state that allows to apply for a passport.

2 Relevant Work

2.1 Personal Information Management

PIM activities can be viewed as “*an effort to establish, use and maintain a mapping between need and information*” [16]. *Need* in that sense depicts a necessity of a task (e.g.: a restaurant reservation before a meeting). To find out when we *need* to be at a meeting we consult a calendar. To find out what we *need* to buy we look at a todo-list. In this work we focus on calendars and todo-lists that often bear a very general and informal representation of tasks or errands. The only machine readable information currently supported are temporal intervals, due dates and to some extent locations. Temporal information is used to trigger alerts, or in some cases checked for overlaps of events. Spatial information is increasingly utilized for location based alerts on mobile devices. But current solutions are rather simplistic and as shown in [22,21,2] the integration of space and time bears more potential for supporting our daily life task planning.

2.2 Affordances and Places

The notion of *affordance* was shaped by Gibson [9], who investigated the perception of the environment. The core assumption is that objects or things are not primarily perceived by discrimination of their properties or qualities, as seen by orthodox psychology. He suggested that humans perceive their environment on the basis of its affordances, hence the possibilities of interaction. A horizontal surface, for example, affords support, what allows walking, as opposed to a steep slope that might afford slipping or falling. So the environment constrains the possibilities of what can be done. Jordan et al. [17] argued for an affordance based model of places, to improve the communication between the GIS and the user. They mention the work of Heft [14] who considers the role of functions or affordances of places in navigational processes. According to them an affordance based model of place is comprised of 6 aspects, listed in Table 1.

Table 1. The defining aspects of an affordance based place model are listed on the left side. The right column shows the implemented representations.

Defining Aspects	Implementation
physical features	location & object collection
actions	pick-up & locomotion
narrative	—
symbolic representations/Names	home,office,shop,etc..
socioeconomic and cultural factors	—
typologies/categorizations	container & street-node

In 2004 Raubal et al. [21] presented a comprehensive theory for location based services (LBS), in which they attempt to combine an *extended theory of affordances* with time geography [13]. It was achieved by embedding affordances into different realms: *physical*, *social-institutional*, and *mental* [20]. The integration of a social-institutional reality into the theory causes the physical affordance of taking a strangers purse, for example, to be suppressed by the context of institutional/social regulations (e.g.: law). Mental affordances are explained as the affordance to decide upon perceived physical or social-institutional affordances possible.

They set the framework for a LBS that can solve scheduling problems for a traveler with different time constraints and preferences. The article illustrates, besides other things, the need for an affordance based place description in order to be able to compute plans or schedules for an agent.

In our solution we adopt a simplified affordance model of place, dealing only with physical features (i.e.: collection of objects available), actions (i.e.: activities possible), categorizations and to some extent names (see Table 1).

2.3 Spatial Behavior, Decision Making and Problem Solving

There are several fields that investigated spatial behavior and decision making extensively. A core issue in transport planning, for example, is the question of service demand. The dominant approaches for modeling it are (1) recording the spatial behavior or (2) the "*examination of the decision-making and choice processes that result in spatially manifested behavior*" [10]. These are referred to as behavioral and structural models. While structural models represent the aggregate movement activities of populations, behavioral approaches try to take the uniqueness of each individual into account. This lead to the investigation of wayfinding as well as cognitive mapping and its impact on spatial behavior. As research has shown, individuals build a cognitive map of their unique mental representation of the world [6]. This information can be facilitated to take decisions about movement in space [18]. This leads to a certain *behavior space* [10] in which our movements are located. Thus, our internal representation of the world has substantial impact on our movements and behavior.

Artificial intelligence is a field concerned with agent behavior, decision making and particular problem solving, but probably in a more formal way. There, a problem can be defined by five components [23]:

- states;
- initial state;
- transitions or actions;
- a goal test;
- path cost.

A solution then is an ordered action sequence that leads from the initial state to the goal state. A famous example for such a problem is the shortest path problem. It describes the question of how to get from one state to another with least effort, assuming a graph like structure. A well known algorithm to solve the problem is *Dijkstra's algorithm* [5]. We will show that by generalization we can narrow down the problem of getting a passport in the fastest possible way, to such a shortest path problem and solve it by the same means. The attempt is to translate the informal task and *behavior space* description into a formal problem description.

3 The Case Study

The scenario we chose to investigate is that of a person who plans to travel abroad and therefore needs a new passport since the old one is expired. In previous work [3] we investigated the planning process undertaken by the agent (see Figure 1). We define the task by two aspects: (1) spatio-temporal requirements, hence the physical presence of the person at the administrative office within the opening hours, and (2) a set of required objects, what we will call *equipment*.

In accordance with the assumptions mentioned in the previous section a *mental/cognitive map* [12,6] is utilized by the agent to map the tasks upon suitable places. For example: in order to acquire a picture the agent is aware of two shops that *afford* the task and are thus included in the decision making process.

The agent further puts the sub-tasks in order, by checking for dependencies. Since the task *photograph acquisition* has money as a precondition, it was derived that *money acquisition* needs to be conducted beforehand.

Finally the tasks are translated into a series of future actions (see bottom of Figure 1). In that part again a mental representation of the world is facilitated for the determination of travel times and routes.

3.1 Problem Definition

Before elaborating on the proposed solution we provide a formal definition of the problem, according to the 5 components listed in section 2.3.

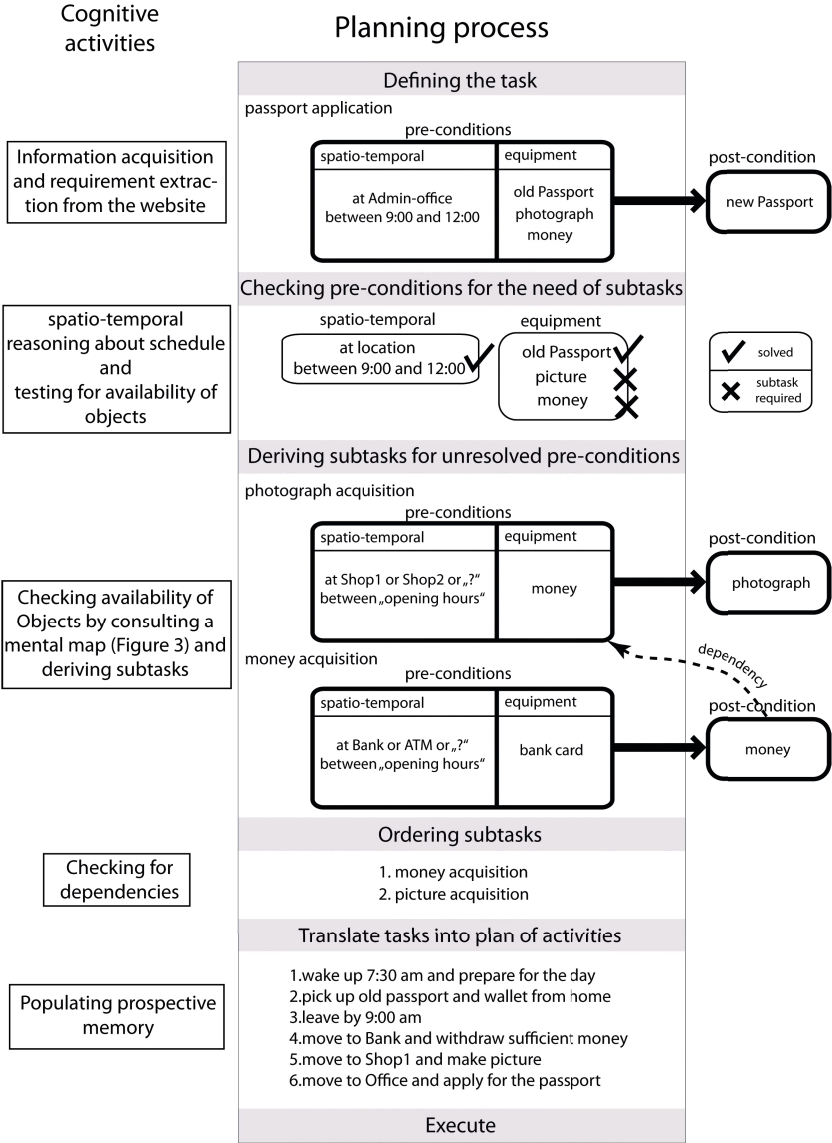


Fig. 1. A simplified illustration of the planning process the person goes through. On the left the person’s cognitive activities are pointed out. In the middle the planning process is subdivided into certain steps. Outcomes for the defined tasks are found on the right side. [3]

States: The problem definition uses a deterministic environment, hence each next state is determined by the transition (i.e.: the action the agent takes) that lead to it. The states in our search space are determined by a product value of

the set of all locations L and the set E of all equipment states (i.e.: the objects carried). The state-set S is therefore defined as:

$$S \subseteq (L \times E)$$

S stands for all possible combinations of locations and equipment states, given the affordance constraints enforced by the environment. S can be seen as the formalization of a *behaviour space*.

Initial State: The initial state for our case study is *home* as location and an empty set of objects as equipment. We denote the start-state s as :

$$\begin{aligned} s &= (l, O) \text{ where } l \in L \text{ and } O \subseteq E \\ &\text{with} \\ l &= \textit{home} \\ O &= \emptyset \end{aligned}$$

Transitions: Transitions or actions represent the rules of how a state can change from one to another. We consider two different kinds of actions (1) locomotion (e.g.: movement from one location to another) and (2) manipulation (e.g.: picking up an object). Each transition has a cost value c attached to it, in this example a value depicting the time an action takes in minutes. A transition is denoted as:

$$t = ((l, O), (l, O), c) \text{ with } l \in L, O \subseteq E \text{ and } c \in \mathbb{R}, c \geq 0$$

Goal Test: The goal is achieved when a solution to the problem with the least cost is found. Thus an ordered sequence of transitions t that leads from the initial state s to the goal state g , with a minimum cost sum.

$$\begin{aligned} g &= (l, O) \text{ where } l \in L \text{ and } O \subseteq E \\ l &= \textit{office} \\ O &= \{\textit{oldpassport}, \textit{photograph}, \textit{money}\} \end{aligned}$$

Path Cost: In order to compute a final cost for a solution we need to be able to build a total sum over all the transitions in a path sequence, independent of the type (i.e.: manipulation or locomotion). For the example we defined the cost to be a temporal value, thus a certain amount of time spent on each action.

4 Solution

The problem definition above is already on a very generalized and abstracted level, but as we learn from figure 1 there are quite a number of cognitive activities involved to get from the general description of the task to a level of abstraction that allows to compute a plan. In the following we propose a formalism that helps

building the state space and transitions defined in 3.1.1 and 3.1.3. We start with the definition of an algebraic model of the relevant environment, out of it we produce a state transition network representing the possible orders of object acquisition and finally combine the two into a single search space. Subsequently this allows us to run a well known problem solving algorithm (i.e.: Dijkstra's shortest path) to determine an optimal solution to the posed task.

The implementation was done in the functional programming language Haskell [25].

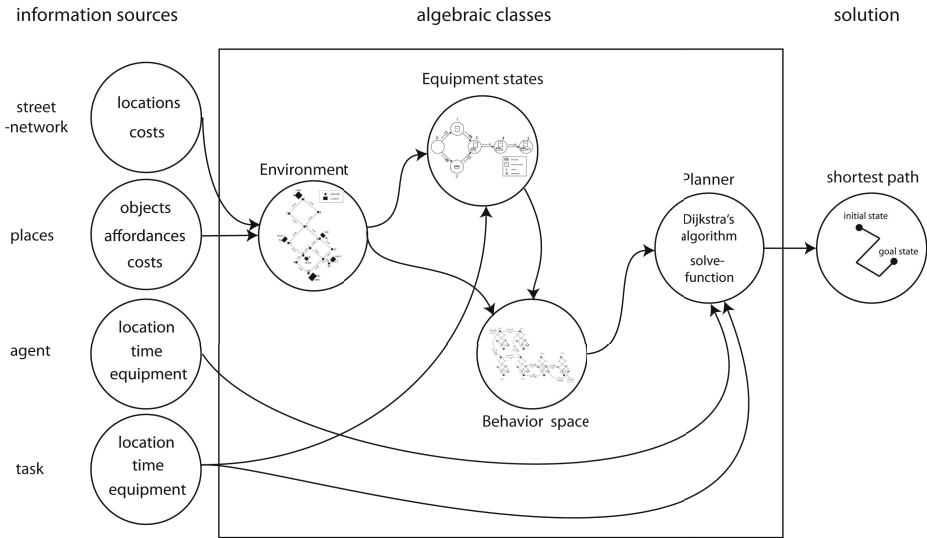


Fig. 2. A graphical illustration of the solution. The information sources on the left represent databases or data structures that form the basis for the algebraic classes.

4.1 Representing the Environment

In the first step we need a representation of the environment, that suffices the needs of an agent to solve the task. We therefore build an algebraic model of the environment sketched in figure 3. It was modeled as a collection of places that we can act upon, based on the affordances it offers and the relations between them. The underlying data structure utilized, is a graph that contains information about places, the objects contained, affordances and their locations. The implementation models some places as image schematic containers [15] and hence distinguish them from simple street junctions that are not necessarily perceived as such. A *container* offers a *pick-up* affordance for objects contained and a *locomotion* affordance to move to neighboring nodes, whereas street junctions offer locomotion only.

```

class Environment graph location object where
  affordance :: graph -> location -> [action]
  locations  :: graph -> [location]
  adjacent   :: graph -> location -> [location]
  travelcost :: graph -> location -> location -> TCost
  pickupcost :: graph -> Object -> location -> TCost
  locomotions :: graph -> [locomotion]
  nextLocation :: graph -> locomotion -> node
  queryObj   :: graph -> Object -> [node]
  queryObjAt :: graph -> location -> requirements
  queryAllObj :: graph -> [Object]

```

The function *affordance* returns a list of activities afforded by a place at a specific location. The second function returns a list of all locations that are accessible to the agent. The *adjacent* function returns all locations that are immediately accessible to the agent by a single *locomotion*, or simply the neighboring nodes for a given node.

To acquire the cost that is assigned to the *locomotion* from one location to another the *travelcost* function is implemented. The *pickupcost* returns the cost that a pick-up action applied to a certain object at a specific location takes.

The *locomotions* function returns a list of all possible transitions/locomotions that can be conducted in the environment. To determine the next location of the agent given a *locomotion* the *nextLocation* function is defined.

The final three functions are concerned with the objects at different locations. While the first returns a list of locations where a specific object can be found, the second returns the preconditions a *pick-up object* affordance exhibits. The requirements come in the form of *equipment*.

The last function simply returns all the objects that can be *picked-up* in the environment.

By having the environment represented in such a way we can in the next step extract a representation of the equipment states, taking the preconditions of *pick-up* object affordances into account.

4.2 Modeling the Requirements

Now we can start building a finite and deterministic state-machine that models the possible equipment-states. Such a state machine in general is defined by a set of states S ¹, a transition function δ and an alphabet A . S is defined as:

¹ Note that the S here represents the equipment-state and thus differs to the S introduced in the problem definition.

$S \subseteq \mathcal{P}(O)$ with O being the set of required objects. Because we only consider directed edges, the subset S can be computed from $\mathcal{P}(O)$ by checking it for *consistency*. It means that an object collection cannot contain money without a bankcard for example, since the bankcard is a precondition to the *pick-up money* affordance. This helps to keep the example small and clear.

The transition function is defined as $\delta : O \times S \rightarrow S$, hence takes an object and a state and returns the next state.

We implemented the state machine as a class (see figure 4 for an illustration):

```
class ObjectAutomata state transition where
  makeStateSet :: Equipment -> [state]
  pickup      :: Object -> state -> state
  makeTransitions :: [state] -> Equipment -> [transition]
  possibleTransition :: state -> [transition]
  nextState    :: state -> transition -> state
```

The first function creates the state set S , taking a set of required objects as an input. The pickup-function denotes the transition function δ by taking an object and a state as an input and returning the resulting state.

The subsequent two functions return all the possible transitions/actions (*makeTransitions*) and all the transitions possible from a specific state (*possibleTransition*).

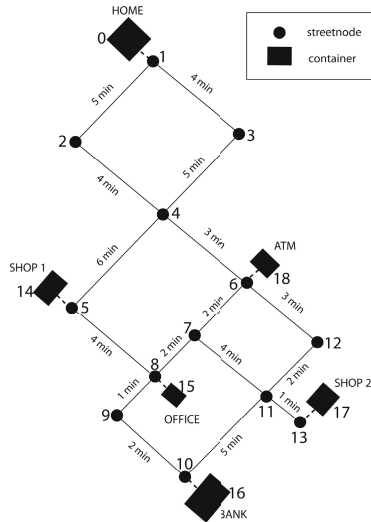


Fig. 3. The environment we consider consists of a street network and places that are linked to it. The places offer a pick-up affordance upon a set of objects. Each locomotion from one streetnode to another is uniquely defined by its start- and endnode and has a cost value attached to it.

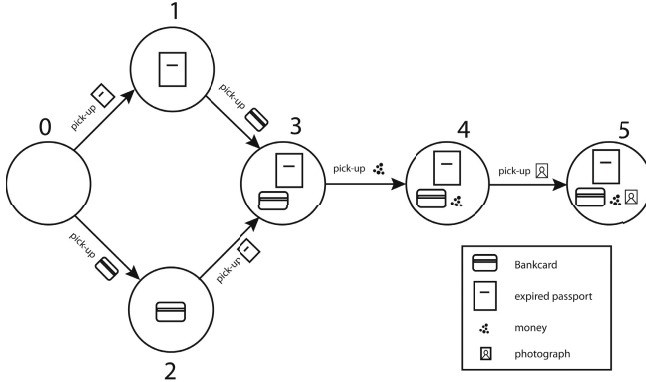


Fig. 4. By using only a single action, we can simplify the graph that depicts the state machine. Each transition is defined by its start- and endstate.

The final function takes a state description and a transition to return the next state. Having a representation of the object order, allows to proceed to the step of combination.

4.3 Combining Environment and Task Requirements

To reach the state space described in section 3.1.1 we need to combine the environmental representation with that of the object requirements. Therefore a methodology suggested by Frank [7] is used. The work describes a mathematical formalism to merge two state-transition networks, based on category theory [4]. The combined state is defined by a pair of the individual location states and equipment states. The combined actions are simply the sum of both costs. To take environmental constraints into account, combination rules are introduced. These define the possible transition from one product-state to another. In our case these rules are determined by the locations that offer *pick-up* affordances for the required objects. Figure 5 gives an illustration of how the combined graph looks like. Again we represent the combined network as a class:

```
class BehaviorSpace combination state where
  states :: combination -> [state]
  transitions :: combination -> [(state,state,TCost)]
  nextTrans :: combination -> state -> [state]
  cost :: [(state,state,TCost)] -> state -> state -> TCost
```

The *combination* input of the first three functions is defined as a vector containing the minimum information needed to build the two different representations (environment and equipment states) described in 4.1 and 4.2. These are on the one hand a data structure or database that contains the information about the environment and on the other the set of objects required for the task.

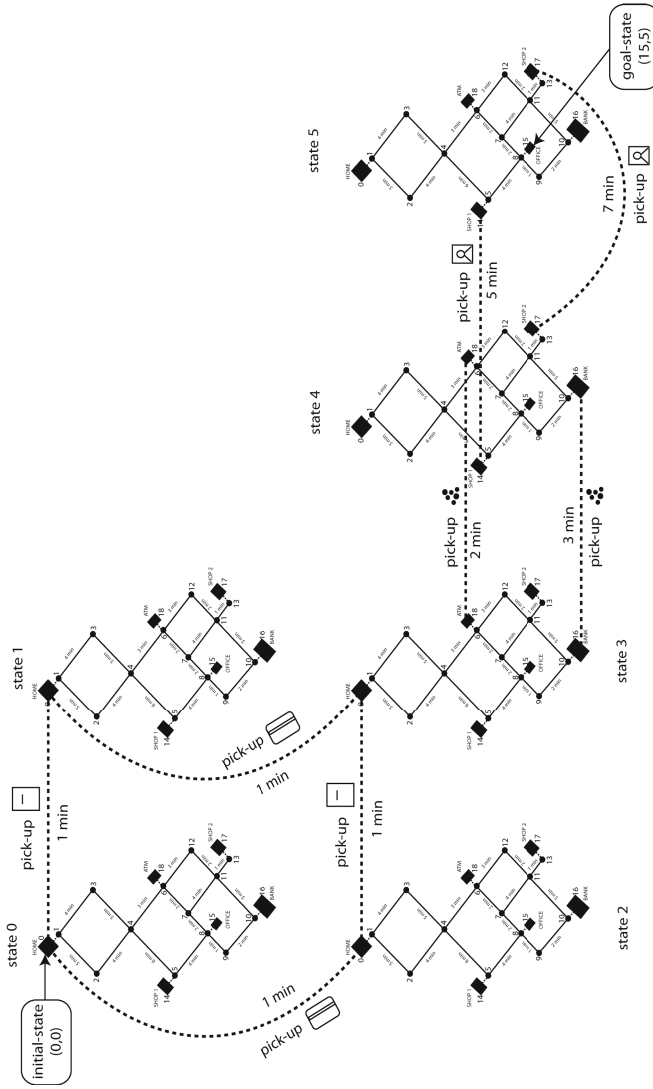


Fig. 5. A combined state is defined as a pair of an individual location state and an equipment state. To switch into the next equipment state a manipulation transition is needed (dotted lines), to move from a location to another a locomotion is required (street edges). The combination rules depict where a manipulation can take place.

The functions defined to describe the *BehaviorSpace* class are conceptually the same as some of the functions defined for the initial environment. Derivable is a set of states, a set of all possible transitions, a set of all the possible transitions based on a specified state and the cost for a given transition.

4.4 Problem Solving

In the final step we created a planning class, that mimics some of the cognitive activities shown in figure 1. The class is defined as follows:

```
class Planner behaviourspace state where
  stp :: behaviourspace -> state -> state -> [(Action, Cost)]
  solve :: behaviourspace -> Agent -> Task -> [(Action, Cost)]
```

The *stp* function computes the shortest path from the current state of an agent to the goal state defined in section 3.1.4, by running a *Dijkstra's algorithm*. To supplement the model we defined an agent and a task description that is used as an input for the problem solving function *solve*:

```
data Agent = Ag Location Time Equipment
data Task = T (Location, TimeInterval) Requirements
agent = Ag home (08,00) []
goal = T (office, ((09,00), (12,00))) [passPhoto, money, oldPassport]
```

Note that the above definitions include time. We did not take temporal aspects into account, although we believe including it to some extent is straightforward. The final function *solve* takes, besides the agent and the task description, the computed *behavior space* and the required *equipment* as an input. The function then compares the agent's current state to the goal state defined in the task. It determines the set of missing objects and passes it to the *stp* function that builds the requirement state transition graph (figure 4), extract the combination rules and combine it with the street network graph. The result is an optimal solution in form of a series of actions, locomotions and manipulations, along with the time it takes to conduct it.

5 Discussion and Outlook

The paper presents the attempt to merge information stored in a PIM-tool (i.e.: calendars or todo-lists) with a typical functionality found in GIS or navigation applications, i.e.: routing. By using a simple example we investigated a formalism that translates it into a *shortest path problem*. Using mathematical category theory, two semantically different state transition networks were combined into a single one and a plan that represents spatial behavior was computed.

The work highlighted some of the issues that arise when trying to combine navigation with ordinary tasks. Foremost the need of an affordance based place model that integrates small scale objects. A GIS aiming at personal tasks, has to

offer a more user centric representation of the environment. For our daily tasks the geometric or map metaphoric information is less important than activities, objects and people. Therefore we introduced the notion of *equipment*, that we believe plays a prominent role in our daily life. Not having a photograph when intending to apply for a passport results in failure.

Another serious issue is the question of complexity. We simplified the model by including two affordances: *pick-up* and *locomotion*. But there are infinitely more actions possible. Thus: what level of granularity is necessary for such an application? Do we need to consider actions like "pay" or "open door" that take place inside a shop?

In a previous paper [1] we envisioned PIM-tools that act pro-actively by alerting us in situations that threaten the success of a task. Therefore we would not just need an optimal solution, but a representation of all possible solutions, enabling it to determine when engagement is necessary. Further the question of how to handle several tasks needs to be investigated.

Further the computation of paths based on other *costs* are possible, hence *cheapest* rather than *fastest* (e.g.: Find the cheapest path for my shopping list!).

The study can be seen as a first step towards the merging of PIM-tools and GIS. Hopefully in future we can solve some of the hindrances discussed, leading to task aware and more intelligent calendars.

References

1. Abdalla, A., Frank, A.U.: Personal Geographic Information Management. In: Proceedings of the Workshop on Cognitive Engineering for Mobile GIS, Belfast, USA. CEUR Workshop Proceedings (2011)
2. Abdalla, A.: Latyourlife: A Geo-Temporal Task Planning Application. In: Advances in Location-Based Services. Lecture Notes in Geoinformation and Cartography, pp. 305–325. Springer, Heidelberg (2012)
3. Abdalla, A., Frank, A.U.: Towards a spatialization of PIM-tools. In: GIZeitgeist 2012: Proceedings of the Young Researchers Forum on Geographic Information Science, Muenster, ifgiPrints (2012)
4. Asperti, A., Longo, G.: Categories, Types and Structures - An Introduction to Category Theory for the Working Computer Scientist, 1st edn. Foundations of Computing. The MIT Press (1991)
5. Dijkstra, E.W.: A note on two problems in connection with graphs. *Numerische Mathematik* 1, 269–271 (1959)
6. Downs, R., Stea, D., et al.: Maps in minds: Reflections on cognitive mapping. Harper & Row, New York (1977)
7. Frank, A.U.: Shortest path in a multi-modal transportation network: Agent simulation in a product of two state-transition networks. *KI Künstliche Intelligenz* 3, 14–18 (2008)
8. Gärling, T., Golledge, R.: Cognitive mapping and spatial decision-making. In: Cognitive Mapping: Past, Present, and Future, p. 47. Routledge, London (2000)
9. Gibson, J.: The Ecological Approach to Visual Perception. Erlbaum Books (1979)
10. Golledge, R.G., Gärling, T.: Spatial behavior in transportation modeling and planning. In: Goulias, K.G. (ed.) *Transportation Systems Planning: Methods and Applications*, pp. 3/1. CRC Press, New York (2003)

11. Golledge, R., Gärling, T.: Cognitive maps and urban travel. In: Handbook of Transport Geography and Spatial Systems, vol. 5, pp. 501–512 (2004)
12. Gould, P., White, R.: Mental Maps. Allen & Unwin, Sabine (1986)
13. Hägerstrand, T.: What about people in regional science? Papers of the Regional Science Association 24, 7–21 (1970)
14. Heft, H.: The ecological approach to navigation: A gibsonian perspective. In: The Construction of Cognitive Maps, pp. 105–132 (1996)
15. Johnson, M.: The Body in the Mind: The Bodily Basis of Meaning, Imagination, and Reason. University of Chicago Press, Source: David Mark (1987)
16. Jones, W., Teevan, J.: Personal information management PIM, 14 p. Univ. of Washington Press (2007)
17. Jordan, T., Raubal, M., Gartrell, B., Egenhofer, M.: An affordance-based model of place in gis. In: 8th Int. Symposium on Spatial Data Handling, SDH, vol. 98, pp. 98–109 (1998)
18. Kuipers, B.: Modeling spatial knowledge. Cognitive Science 2(2), 129–154 (1978)
19. Lepouras, G., Dix, A., Katifori, A., Catarci, T., Habegger, B., Poggi, A., Ioannidis, Y.: Ontopim: From personal information management to task information management. In: Personal Information Management: Now That We are Talking, What Are We Learning?, p. 78 (2006)
20. Raubal, M.: Ontology and epistemology for agent-based wayfinding simulation. International Journal of Geographical Information Science 15(7), 653–665 (2001)
21. Raubal, M., Miller, H., Bridwell, S.: User-centred time geography for location-based services. Geografiska Annaler: Series B, Human Geography 86(4), 245–265 (2004)
22. Raubal, M., Winter, S., Teÿmann, S., Gaisbauer, C.: Time geography for ad-hoc shared-ride trip planning in mobile geosensor networks. ISPRS Journal of Photogrammetry and Remote Sensing 62(5), 366–381 (2007)
23. Russell, S., Norvig, P.: Artificial intelligence: a modern approach, pp. 3/66. Prentice-Hall (2010)
24. Stern, E., Portugali, J.: Environmental Cognition and Decision Making in Urban Navigation. In: Wayfinding Behavior: Cognitive Mapping and Other Spatial Processes, pp. 99–118. John Hopkins Press (1999)
25. Thompson, S.: Haskell - The Craft of Functional Programming. International Computer Science Series. Addison-Wesley (1996)
26. Timpf, S., Volta, G., Pollock, D., Egenhofer, M.: A Conceptual Model of Wayfinding Using Multiple Levels of Abstraction. In: Frank, A.U., Formentini, U., Campari, I. (eds.) GIS 1992. LNCS, vol. 639, pp. 348–367. Springer, Heidelberg (1992)

Automated Centerline Delineation to Enrich the National Hydrography Dataset

Chris Anderson-Tarver¹, Mike Gleason¹, Barbara Buttenfield¹,
and Larry Stanislawski²

¹ Dept. of Geography, University of Colorado-Boulder, Guggenheim 110, 260 UCB Boulder,
Colorado 80309, USA

{anderson, michael.j.gleason, babs}@colorado.edu

² Center of Excellence for Geospatial Information Science (CEGIS), United States Geological
Survey (USGS), Rolla, Missouri USA

lstan@usgs.gov

Abstract. A common problem in the automated generalization of basemaps is extraction of important features for cartographic visualization purposes. The delineation of a stream network centerline poses unique challenges especially when variables such as stream order, channel depth, or flow rate are not available. This paper presents an algorithm for automated delineation of a continuous cartographic centerline through a flowline network encompassing a single subbasin. Six datasets testing the algorithm are drawn from the U.S. National Hydrography Dataset (NHD) to compare among delineations in landscapes with varying terrain and precipitation regimes. Centerline delineation provides a database enrichment, which adds functionality and enables cartographic generalization. A user-defined cutoff value permits progressively inclusive centerline delineations which may be targeted to multiple map scales and purposes.

Keywords: Stream centerline, cartographic generalization, database enrichment.

1 Introduction

During the map design process, the abstraction of important features given a specific cartographic purpose is a typical task. Abstraction is described by Buttenfield and Mark [6] consists of information processing through generalization or information encoding through symbolization. The selection of crucial characteristics during generalization is dependent on map purpose and target scale [5]. Symbolization is also bound by conceptual constraints within a given map design [6] and, therefore, clear and specific attribute definitions are mandatory.

Since feature abstraction is partially limited by encoded attributes as well as by the conceptual definition, enrichment procedures can augment the database in preparation for map design. Enrichment is especially important in vector databases where prioritizing feature importance can be a necessary, but time-consuming task [13]. When accomplished manually, feature prioritizations are sometimes inconsistent or

incomplete. Enrichment procedures add explicit attribution to a database and need to be identified prior to finalizing the map design [9]. Neun et al [14] enrich thematic data to support adaptive generalization. The same researchers [13] argue that enrichment will "... equip the raw spatial data with additional information about objects and relationships" (p.1), and that enrichment may support data characterization, conflict detection, parameter selection, algorithm choice, and processing evaluation. They invoke web-based support services to enrich cartographic data with priority sequencing or adjacency relations, which can facilitate subsequent generalization operators as for example object aggregation.

The value of enrichment processing has been described in applications relevant to generalization. Balboa and Lopez [2] use Principal Components Analysis to select characteristics to enrich a transportation network for automatic classification of road types. Their stated conditions for effective enrichment include that the set of characteristics be statistically significant, uncorrelated, and normalized. Steiniger and Weibel [20] enrich cartographic data by storing vertical and horizontal relations among data items, to reduce computational and predicate workloads in subsequent processing. Enrichment has been applied to evaluate shape preservation following simplification and realignment of buildings [25]. Data enrichment includes two tasks: first to model implicit relations within the data, and second to add these relations explicitly to the database in a form that is accessible to subsequent generalization operators [4].

Centerlines provide an excellent problem domain for enrichment. Cartographers highlight stream centerlines to identify the most important water path through the network. Enrichment by attributing centerlines can assist with symbolization hierarchies, as stream labels are often attached to the cartographic centerline but not to flowlines of lesser importance. At smaller scales, the cartographic centerline may be the only feature depicted for a network, essentially comprising the last remaining water channels following simplification. Previous work by the authors [1] implemented a centerline extraction algorithm which the present paper extends. The earlier version began by selecting flowline features attributed as artificial paths [23], which represent a flow path through polygonal water features, such as wide rivers and lakes. Artificial paths were assumed to constitute the main channels of the subbasin since large channels are often wide enough to be captured as polygons. Gaps between artificial paths were filled using a cursor search to identify flowline segments with common nodes. To speed the cursor search, the algorithm looked for groups of flowlines whose reachcode feature identifier occurred in an arithmetic sequence, assuming that numerically similar reachcodes would be geographically proximal.

Whereas this approach was initially considered successful, further testing revealed that the numeric proximity assumption does not hold true in many subbasins. Also, the use of artificial paths created a centerline with too many branches, especially in areas with extremely wide channels. Another limitation was discovered in that subbasins in dry landscapes often lack standing water polygons or channels large enough to be captured as polygons, and thus, artificial paths could not be used as a starting point for the centerline framework. Finally, the algorithm did not ensure full extension of the delineated centerline from headwaters to pour point.

This paper redesigns and extends the work reported in [1]. The innovation in the current work is to support centerline delineation on the basis of estimated upstream drainage area (UDA) values. Enriching the data with UDA allows for a relative ranking of stream reaches based on cumulative catchment area, and has proven utility in any hydrographic database that lacks explicit stream order attribution [18]. The use of UDA values constitutes a fundamental difference to the approach described in [1], in several respects. Catchment areas can be estimated for each line feature, thus eliminating the possibility of encountering gaps. The revised algorithm does not depend on the presence of artificial paths or numerically similar reachcodes. The new algorithm supports delineation of a continuous centerline extending completely across one or multiple hydrographic subbasins. Also, the new implementation reduces the need for cursor searching, which speeds centerline delineation considerably. The paper will present the algorithm and show results for six study areas, to show its performance in varying types of landscapes. The significance of tailoring generalization processing to specific landscape types has been demonstrated previously [7, 8, 16, 22].

2 Context for a Cartographic Centerline

Cartographic centerlines can be defined as the main channel or channels that run continuously through a subbasin. The main channel might be defined in several contexts, for example as the set of channels containing the largest volume of water, or the set of channels running with maximum velocity. Other disciplines use different criteria and definitions to prioritize the most significant channels in a stream network. Hydrologists for example refer to a stream *main stem* or trunk as part of a hierarchy of tributaries, defining the main stem as "... the primary downstream segment of a river, as contrasted to its tributaries" [3]. Horton's [11] stream ordering system assigned the highest stream order to the complete main stem, although subsequent modifications (such as Strahler [21]) assigned highest order only to the portion downstream of the tributaries with the second highest order. In contrast, fluvial geomorphologists rely on terrain to identify the highest priority stream channel, called a *thalweg*, which is the path joining the points of lowest bed elevation along the river channel [12]. The *thalweg* is commonly referred to in hydraulic modeling as defining the line of fastest river flow [10].

In this paper, the specification of a centerline is based on the channels draining the largest area within the subbasin. The area of upstream drainage is a significant factor used to estimate stream flow in the National Flood Frequency program [24] and will be used for the research reported below.

3 Datasets and Study Areas

The National Hydrography Dataset (NHD) is a vector database of surface-water features of the United States maintained and coordinated by the U. S. Geological Survey (USGS). Multiple resolutions of NHD include a Medium Resolution (MR),

compiled for use at 1:100,000 and smaller mapping scales, and a High Resolution (HR), which was initially compiled for use at 1:24,000 in the coterminous states and 1:63,360 in Alaska. A third, Local Resolution (LR) database includes HR features densified to 1:2,400 scale and is available in only a few areas of the Nation as needed by state and local agencies.

This paper will focus on the HR data. As discussed, stream order could provide a basis for centerline delineation (http://nhd.usgs.gov/nhd_faq.html). However, because of the large geographic extent and data volume, as well as the frequent and irregular update cycles undertaken through the NHD stewardship program, USGS does not maintain stream order information for HR flowlines. Currently, the Center of Excellence for Geospatial Information Science (CEGIS) and the Meridian Lab at University of Colorado-Boulder are investigating methods to enrich the NHD for use in cartographic base mapping and generalization in support of *The National Map*. A part of this research investigates automated centerline delineation approaches that are flexible enough to delineate centerlines in the absence of explicit stream order.

Centerlines may of course differ in the number of main channels per subbasin, or the number of branches per main channel. A centerline may have minimal to extensive braiding, and or may contain discontinuities such as underground streams. The algorithm set forth in this paper is tested on six subbasins from three different terrains and two different precipitation regimes [7, 8, 19], illustrating a wide range of physiographic types across the coterminous United States [7]. Table 1 lists the six HR subbasins examined in this paper and their associated landscape types.

Table 1. Six NHD High Resolution subbasins demonstrating prototypical landscape type important for evaluating enrichment research in support of *The National Map*

NHD subbasin name, location	Subbasin ID	Landscape Type
Upper Suwanee River, Georgia-Florida	03110201	Flat Humid
Lower Beaver River, Utah	16030008	Flat Dry
Pomme de Terre, Missouri	10290107	Hilly Humid
Lower Prairie Dog Town Fork Red River, Texas	11120105	Hilly Dry
South Branch Potomac, West Virginia	02070001	Mountainous Humid
Piceance and Yellow Creeks, Colorado	11050003	Mountainous Dry

Enrichment tools are being developed in support of *The National Map* as part of a continued effort to design a comprehensive approach for generating reduced scale versions of NHD. The role of NHD enrichment in this project is to facilitate generalization by automated pruning, which removes entire confluence-to-confluence stream segments, and simplification, which eliminates coordinates from remaining stream segments (Fig. 1). NHD enrichment includes classification of land types and partitioning of stream channel densities for tailoring of subsequent pruning and simplification routines. Additional enrichment measures include attributing each flowline with catchment area and upstream drainage area (UDA) estimates, to enable pruning for generalization and support cartographic purposes such as symbolizing tapered stream lines, as well as informing the centerline delineation process, as described below.

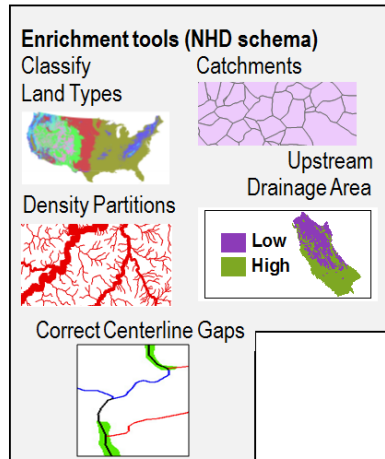


Fig. 1. Enrichment tools in development for the NHD Generalization Toolbox. Each tool generates attributes not currently provided within the standard NHD schema; and these attributes direct subsequent generalization, including feature pruning, simplification, and sequencing and types of processing

Several constraints are placed upon enrichment, namely that the procedures be automated through software that can readily access The National Map data format; and that the tools and generalized data versions are distributed in the public domain. The centerline tool carries additional specific constraints. Centerline delineation must be based on geometric characteristics of subbasins and not on identifiers that may change with subsequent update cycles within the NHD. The algorithm must not corrupt stream confluence topology, displace stream channels or add new flowlines to the stream network. The current version of the algorithm is driven by a user-defined UDA cutoff value, which permits a flexible delineation of primary channels.

4 Methods: Algorithm Design

4.1 Preprocessing Steps

The centerline algorithm uses a critical preprocessing step by assigning UDA estimates to each NHD flowline feature. UDA values are estimated through a graph traversal that accumulates catchment areas estimated for each flowline by means of Thiessen polygons [18]. The catchment area and UDA estimation are not intrinsic to the standard NHD schema in current use. The subbasin boundary defines catchments for features abutting the edge of the subbasin, and the area of the subbasin polygon represents the total catchment area for flowline features within the subbasin. Fig. 2 shows catchments generated from Thiessen polygons and associated flowline UDA values for the Piceance-Yellow subbasin in Colorado.

Enrichment processing is regularly performed on single-subbasin datasets. In many subbasins, streams from an adjacent subbasin flow into the flowline network of the target subbasin. In these cases, an obviously large UDA value of one million square kilometers (sq. km.) is added to each feature having an inflow from an adjacent subbasin. Consequently, all features downstream from a feature with an incoming UDA value will be greater than one million sq. km. Adding an inflowing UDA value in this manner is advantageous because it normalizes downstream subbasin drainage areas relative to upstream subbasins, and normalizes UDA values in preparation for expanding the scope of UDA enrichment beyond one or a small number of subbasins.

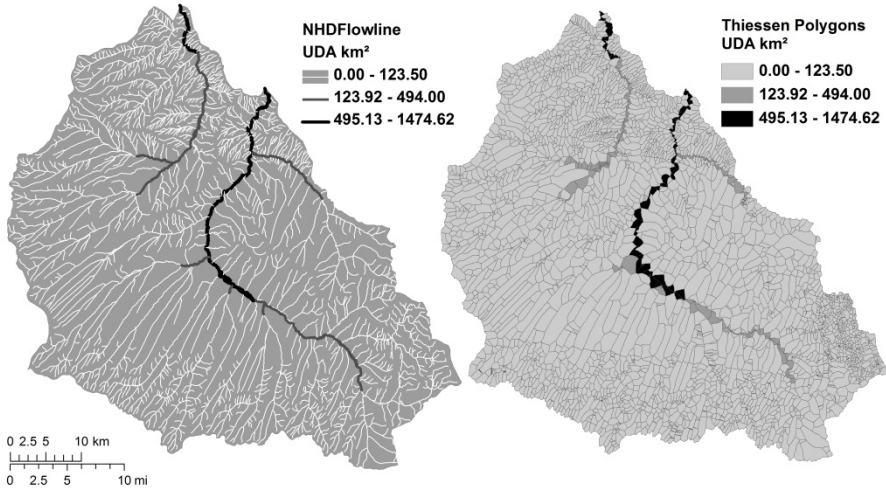


Fig. 2. Estimated UDA and Thiessen polygons values for flowlines in the Piceance-Yellow subbasin in Colorado (NHD subbasin # 14050006), using a method by Stanislawski et al [18]. In the left map, each stream flowline is given a UDA value after it is encased by a single Thiessen polygon, depicted in the map on the right, to estimate the local catchment area for each confluence-to-confluence channel. Areas are tallied upstream to the headwaters for each channel to compute the estimate of UDA.

4.2 Centerline Delineation Algorithm

The centerline algorithm starts with a selection of flowlines based on a user-defined threshold UDA value. To maintain comparability when processing multiple subbasins, the threshold value is input as a percentage of the subbasin area drained by a given candidate. Flowlines whose UDA value is greater than the threshold drain a larger percentage of the subbasin, and are added to a list of centerline “stems”. Thus the cartographic centerline is formally defined by this algorithm as the set of flowlines that run continuously from headwaters to pour point in the subbasin and that drain a substantial percentage of the subbasin area. The algorithm must accomplish two

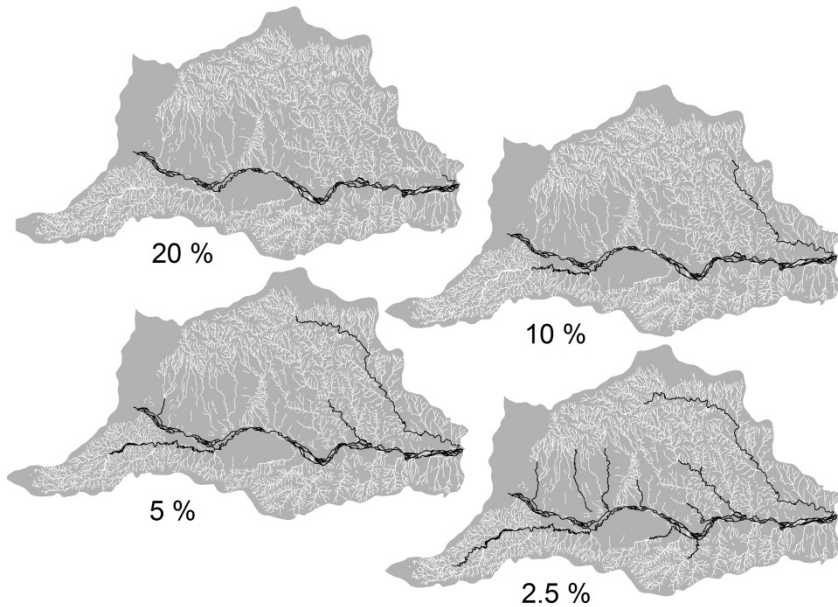


Fig. 3. Comparison of centerline stems resulting from four progressively inclusive Upstream Drainage Area (UDA) cutoff values for the Lower Prairie Dog Town Red subbasin (NHD subbasin # 11120105). Discontinuities in the centerline stems can result from stream segments which flow underground, or from the presence of NHD waterbody or area polygons, which are not displayed in the figure.

major tasks, first to identify a set of centerline stems and second, to trace upstream from this list to the headwaters, insuring that each centerline stem is extended to trace a continuous path upstream to subbasin headwaters (Fig. 3).

In testing, a UDA cutoff value of 20% of the subbasin area selected a sufficient centerline framework for all six test subbasins. Sufficiency in this case means that the number of selected stems in each subbasin encapsulated the primary channel or channels, thus completing the first task, in preparation for the second task of the algorithm (tracing each stem to its headwaters). Fig. 4 reveals the proportion of flowline features in a subbasin collected as centerline candidates for a 20 % UDA cutoff in the Piceance-Yellow subbasin.

Fig. 2 and Fig. 4 show how these candidates give a conservative estimate of the centerline: In Fig. 2 the black features of the left map represent centerline stems for a 20% cutoff value. This cutoff initializes a set of stems for more than one primary channel, but does not include too many tributaries in the stem list. (We do not imply that 20% should be the default cutoff in every case, rather that it provides a workable value for testing the prototype code. Further research will be needed to establish guidelines for appropriate cutoff selection. This point will be raised in the final section of the paper.)

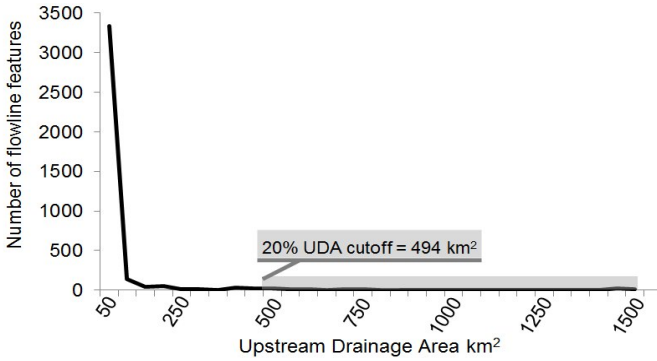


Fig. 4. Histogram of Piceance-Yellow subbasin UDA values demonstrating how many flowlines are added to the candidate main stem list for a 20% UDA cutoff selection (grey box). The leftmost bin of the x-axis represents an increment of 0-50. A 20% UDA cutoff value selects 108 features. The cutoff value of 494 km² is found by multiplying 20% by the subbasin area.

To address the second task, the algorithm traces a single path upstream from each stem to its headwaters. This process begins by finding the upstream endpoint from each centerline stem, and then searches to find all flowline segments attached to and immediately upstream from that point. If a single segment is found, it is added to the centerline stem. If multiple upstream segments are found (e.g., at a tributary confluence), the segment with the higher UDA is selected and added to the stem and the other segment is ignored. This process repeats iteratively until the edge of the subbasin is reached, and no additional upstream segments can be found. The upstream trace then repeats for all other stem endpoints. Once each centerline is traced from subbasin mouth to headwaters, each stem flowline is enriched with an attribute designating it as a centerline.

5 Results

5.1 Delineation of Centerlines

The algorithm was implemented in Python 2.6.5 using the ArcPy module from ArcGIS 10. Performance of the upstream trace is greatly enhanced by the use of native ArcPy and Python objects. Each flowline segment is extracted to a tuple containing its associated polyline geometry object and UDA value. Tuples are grouped into lists for the stems and for the remaining flowlines. Storing the geometry and UDA in tuples minimizes input/output performance slowdown associated with using cursor objects. Each group of tuples may be quickly sorted by UDA value to find the flow-line segment with the maximum drainage.

Each polyline object encapsulates a point object's "from node" (upstream end) and "to node" (downstream end). Access to these point objects allows efficient upstream tracing within the subbasin. For example, to identify the endpoint of each centerline

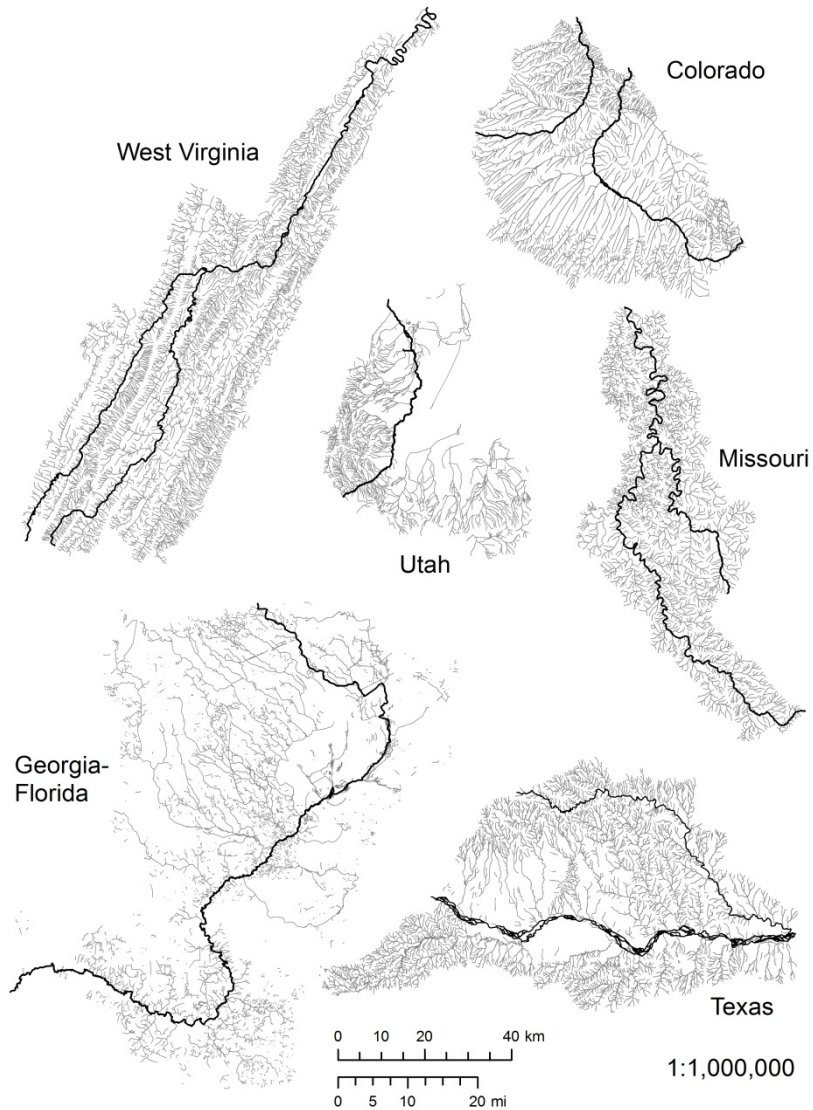


Fig. 5. Centerline delineation results for six NHD subbasins using a cutoff value of 20% of subbasin area

stem, the algorithm simply searches the stem list for all “from nodes” that lack a coincident “to node.” And when tracing through the candidate list, upstream segments are found by searching for “to nodes” that match a previously selected “from node.”

The algorithm builds a master list of tuples representing the complete subbasin centerline(s), and ultimately uses the polyline geometry objects in the master list to select the polylines in the NHD flowline feature class to be attributed as centerlines. Fig. 5 compares centerlines delineated for the six subbasins using a 20 % UDA cutoff.

The three dry landscapes (Colorado, Utah and Texas) contain little standing water, meaning that in these three subbasins, the initial version of the algorithm based on selection of artificial paths failed to generate a centerline at all.

In humid landscapes such as the Missouri subbasin, where the earlier algorithm tended to collect extraneous tributaries (Fig. 6), one can see that the current solution remedies this problem, while maintaining a continuous centerline solution. The new algorithm delineates channels from mouth to headwaters, while ensuring that no gaps exist within the delineation. Because the revised centerline selection relies on UDA instead of artificial paths, the new algorithm provides a simpler, less cluttered channel, which is more appropriate for base mapping.

To produce the results for the Pomme de Terre, Missouri subbasin (Fig. 6), a PC was used with quadcores running at 2.1 GHz clock speed under the Windows 7 operating system. The algorithm ran for 3.22 minutes. This compares favorably to the previous algorithm which took 9.77 minutes to complete its solution on the same PC.

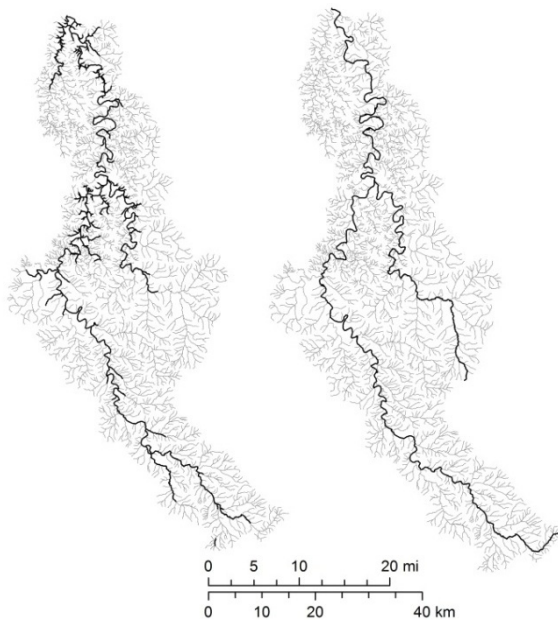


Fig. 6. Comparison of the earlier centerline delineation (left panel) based on artificial paths for the Pomme de Terre subbasin in Missouri, and the current delineation (right panel) based on UDA cutoff values. The earlier delineation reflects a large number of stream tributaries selected in addition to the actual cartographic centerline.

5.2 Validation

Centerline results were validated against stream-order values in a benchmark dataset for the West Virginia subbasin. The NHDPlus MR (1:100,000) dataset for this subbasin was downloaded and the highest order of streams was selected based on a

modified Strahler stream order algorithm called Strahler Calculator, developed for the NHDPlus [15]. Comparison of the West Virginia centerline to the NHDPlus 5th order main stem (Fig. 7A) was performed through a Coefficient of Line Correspondence (CLC), which measures conflation as a proportion of the length of matching stream channels to the total length of channels in the subbasin plus the length of omitted features in the benchmark dataset [17]. A CLC value of 1.0 indicates perfect feature correspondence, whereas a CLC of 0.0 indicates a total mismatch. CLC metrics are useful as they allow for the comparison of features compiled at different scales, and in this case, for slightly different purposes (centerline vs. main stem). The two datasets are overlaid with a 200-cell grid to compute CLC metrics locally across the study area (Fig. 7B), as well as averaging the 200 values for the subbasin as a whole. For West Virginia, the overall CLC metric is 0.632. This value shows a significant difference between the 1:24,000-scale UDA-based main stem and 1:100,000-scale stream-order-based main stem. An extra channel was delineated by the UDA-based algorithm that does not match the 5th order streams in the benchmark. Comparison with 4th order streams produces many additional main stems in the benchmark data, and a similarly low CLC value. The above CLC analysis suggests that the 1:24,000 UDA-based prominence estimate relative to the 1:100,000 5th-order prominence estimate seems a better choice because of improved accuracy and reduced likelihood of omitting a prominent tributary.

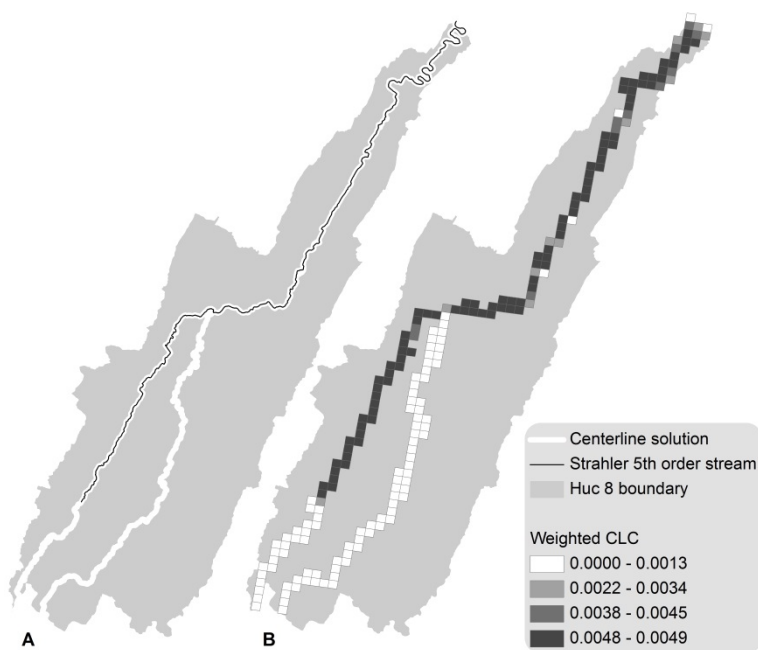


Fig. 7. 20% UDA Centerline Delineation and the 5th order Strahler Calculator [15] in the Pomme de Terre subbasin(A) and Weighted CLC values for 200 grid cells underlying the centerline (B)

6 Summary and Discussion

This paper presents an algorithm for automatically delineating cartographic centerlines, which draws on previous work [1]. The algorithm selects flowlines on the basis of estimated upstream drainage area, and traces common nodes upstream until a headwater is reached. The algorithm makes use of the topologic connectivity of the flowlines and does not alter it in any way. The operation enriches the hydrographic database in preparation for subsequent simplification for base mapping at smaller scales. Parameterizing the UDA cutoff value can furnish a range of centerline delineations, which are more inclusive or more restrictive, according to map purpose and target scale.

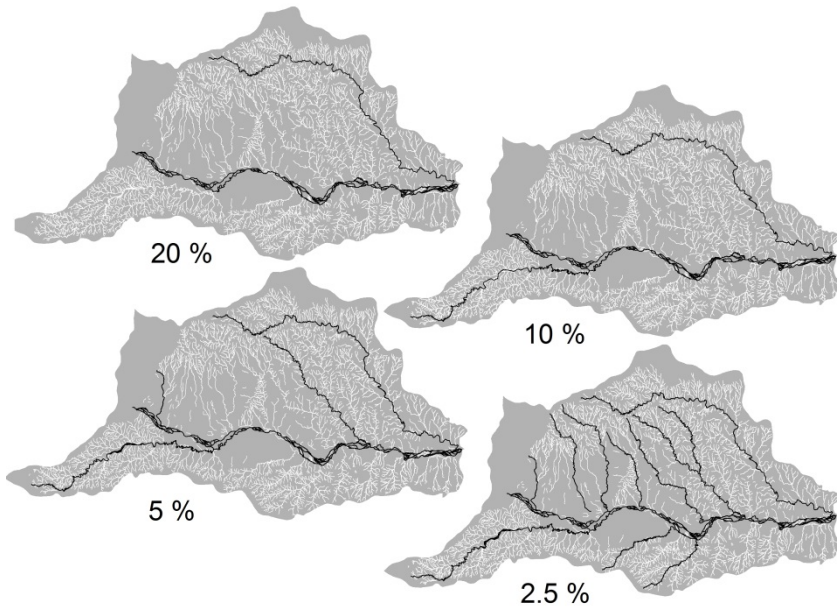


Fig. 8. Comparison of 4 different centerline delineations each starting with different UDA cutoff values for the Lower Prairie Dog Town Red subbasin

The algorithm can be improved in several regards. First, the algorithm works only if stream topology is clean. It requires that flowlines are digitized in the direction of flow, and that all flowlines are attributed explicitly with flow direction. The current version assumes that a user can select a reasonable UDA cutoff value without additional guidance. Selection of tolerance values continues to challenge many aspects of automated generalization. Another limitation is the inclusion of braided stream channels in the centerline delineation. For cartographic base mapping, it is optimal to establish a single primary channel running through each braided area. As Fig. 8 demonstrates, the algorithm currently includes all braided channels whose

UDA value is above the cutoff threshold. Identification of flow hierarchy through braids would provide an alternative to support hydrologic simplification, and this area for further research is currently underway.

Acknowledgements. The work of the Colorado authors is supported by USGS-CEGIS grant # 04121HS029, “Generalization and Data Modeling for New Generation Topographic Mapping”. This paper benefited from the insightful comments of reviewers.

References

1. Anderson-Tarver, C., Buttenfield, B.P., Stanislawski, L.V., Koontz, J.M.: Automated Delineation of Stream Centerlines for the USGS National Hydrography Dataset. In: Ruas, A. (ed.) *Advances in Cartography and GIScience*, Paris. Lecture Notes in Geoinformation and Cartography, vol. 1, pp. 409–423. Springer, Heidelberg (2011)
2. Balboa, J.L.G., López, F.J.A.: Generalization-Oriented Road Line Classification by Means of an Artificial Neural Network. *Geoinformatica* 12, 289–312 (2008)
3. Benke, A.C., Cushing, C.E. (eds.): *Rivers of North America*. Elsevier Academic Press, Burlington (2005)
4. Bobzien, M., Burghardt, D., Neun, M., Weibel, R.: Multi-Representation Databases With Explicitly Modeled Horizontal, Vertical and Update Relations. In: *Proceedings AutoCarto*, Vancouver (2006)
5. Brassel, K.E., Weibel, R.: A Review and Conceptual Framework of Automated Map Generalization. *Int’l Journal of Geographical Information Systems* 2(3), 229–244 (1988)
6. Buttenfield, B.P., Mark, D.: Expert Systems in Cartographic Design. In: *Geographic Information Systems: the Microcomputer and Modern Cartography*, pp. 129–150. Pergamon Press, Oxford (1991)
7. Buttenfield, B.P., Stanislawski, L.V., Brewer, C.A.: Multiscale Representations of Water: Tailoring Generalization Sequences to Specific Physiographic Regimes. *GIScience Short Paper Proceedings* (2010)
8. Buttenfield, B.P., Stanislawski, L.V., Brewer, C.A.: Adapting Generalization Tools to Physiographic Diversity for the United States National Hydrography Dataset. *Cartography and Geographic Information Science* 38(3), 289–301 (2011)
9. Chaudhry, O.Z., Mackaness, W.A.: Automatic Identification of Urban Settlement Boundaries for Multiple Representation Databases. *Computers Environment and Urban Systems* 32(2), 95–109 (2008)
10. Crowder, D.W., Diplas, P.: Using Two-Dimensional Hydrodynamic Models at the Scales of Ecological Importance. *Journal of Hydrology* 230(3-4), 172–191 (2000)
11. Horton, R.E.: Erosional Development of Streams and Their Drainage Basins: Hydrophysical Approach to Quantitative Morphology. *Geological Society of America Bulletin* 56(3), 275–370 (1945)
12. Merwade, V.M., Maidment, D.R., Hodges, B.R.: Geospatial Representation of River Channels. *Journal of Hydrologic Engineering* 10(3), 243–251 (2005)
13. Neun, M., Burghardt, D., Weibel, R.: Web Service Approaches for Providing Enriched Data Structures to Generalisation Operators. *International Journal of Geographical Information Science* 22(2), 133–165 (2008)

14. Neun, M., Weibel, R., Burghardt, D.: Data Enrichment for Adaptive Generalisation. In: Proceedings of the ICA Workshop on Generalization and Multiple Representations, Leicester, U.K., 6 p. (2004), <http://aci.ign.fr/Leicester/paper/Neun-v2-ICAWorkshop.pdf>
15. Pierson, S.M., Rosenbaum, B.J., McKay, L.D., Dewald, T.G.: Strahler Stream Order and Strahler Calculator Values in NHDPlus. Unites States Geological Survey (2008), ftp://ftp.horizon-systems.com/NHDPlusExtensions/SOSC/SOSC_technical_paper.pdf
16. Stanislawski, L.V., Buttenfield, B.P., Finn, M.: Integrating Hydrographic Generalization over Multiple Physiographic Regimes. In: Proceedings of the Symposium on Generalization and Data Integration (GDI 2010) Boulder (forthcoming, 2010)
17. Stanislawski, L.V., Buttenfield, B.P., Samaranyake, V.A.: Automated Metric Assessment of Hydrographic Feature Generalization Through Bootstrapping. In: Proceedings of the 13th Workshop of ICA Commission on Generalisation and Multiple Representations (2010), http://ica.ign.fr/2010_Zurich/genemr2010_submission_11.pdf
18. Stanislawski, L.V., Finn, M., Starbuck, M., Usery, E.L., Turley, P.: Estimation of Accumulated Upstream Drainage Values in Braided Streams Using Augmented Directed Graphs. In: Proceedings AutoCarto, Vancouver (2006)
19. Stanislawski, L.V., Finn, M., Usery, E.L., Barnes, M.: Assessment of a Rapid Approach for Estimating Catchment Areas for Surface Drainage Lines. In: Proceedings ACSM-IPLSA-MSPS, St. Louis (2007)
20. Steiniger, S., Weibel, R.: Relations Among Map Objects in Cartographic Generalization. *Cartography and Geographic Information Science* 34(3), 175–197 (2007)
21. Strahler, A.N.: Quantitative Analysis of Watershed Geomorphology. *Transactions of the American Geophysical Union* 8(6), 913–920 (1957)
22. Touya, G.: First Thoughts for the Orchestration of Generalisation Methods on Heterogeneous Landscapes. In: Proceedings of the ICA Workshop on Generalizations, Montpellier (2008)
23. USGS: The National Hydrography Dataset: Concepts and Contents, United States Geological Survey (2000), http://nhd.usgs.gov/chapter1/chp1_data_users_guide.pdf
24. USGS: The National Flood Frequency Program, version 3: A Computer Program for Estimating Magnitude of Flood for Ungaged Sites, Unites States Geological Survey (2002), <http://pubs.usgs.gov/wri/wri024168/#pdf>
25. Zhang, X., Stoter, J., Ai, T., Kraak, M.J.: Formalization and Data Enrichment for Automated Evaluation of Building Pattern Preservation. In: Proceedings of the Spatial Data Handling 2010, vol. 38(2), pp. 267–273 (2010)

Evolution Strategies for Optimizing Rectangular Cartograms

Kevin Buchin¹, Bettina Speckmann^{1,*}, and Sander Verdonschot²

¹ Department of Mathematics and Computing Science, TU Eindhoven, Eindhoven, The Netherlands

k.a.buchin@tue.nl, speckman@win.tue.nl

² School of Computer Science, Carleton University, Ottawa, Canada
sander@cg.scs.carleton.ca

Abstract. A rectangular cartogram is a type of map where every region is a rectangle. The size of the rectangles is chosen such that their areas represent a geographic variable such as population or GDP. In recent years several algorithms for the automated construction of rectangular cartograms have been proposed, some of which are based on rectangular duals of the dual graph of the input map. In this paper we present a new approach to efficiently search within the exponentially large space of all possible rectangular duals. We employ evolution strategies that find rectangular duals which can be used for rectangular cartograms with correct adjacencies and (close to) zero cartographic error. This is a considerable improvement upon previous methods that have to either relax adjacency requirements or deal with larger errors. We present extensive experimental results for a large variety of data sets.

Keywords: Rectangular cartogram, evolution strategy, regular edge labeling.

1 Introduction

Cartograms [3,13], also called *value-by-area maps*, are a useful and intuitive tool to visualize statistical data about a set of regions like countries, states, or counties. The size (area) of a region in a cartogram corresponds to a particular geographic variable. A common variable is population: in a population cartogram, the sizes of the regions are proportional to their population. The sizes of the regions in a cartogram are not the true sizes and hence the regions

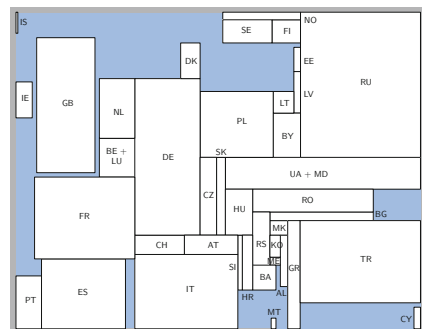


Fig. 1. The population of Europe 2011

* Supported by the Netherlands Organisation for Scientific Research (NWO) under project no. 639.022.707.

generally cannot keep both their shape and their adjacencies. A good cartogram, however, preserves the recognizability in some way.

Globally speaking, there are four types of cartogram. The standard type—also referred to as contiguous area cartogram—has deformed regions so that the desired sizes can be obtained and the adjacencies kept. The most prominent algorithm for such cartograms was developed by Gastner and Newman [8]. The second type of cartogram is the non-contiguous area cartogram [14]. The regions have the true shape, but are scaled down and generally do not touch anymore. Sometimes the scaled-down regions are shown on top of the original regions. A third type of cartogram was introduced by Dorling [4] and is in its original form based on circles. Dorling cartograms maintain neither correct adjacencies between regions nor correct relative positions. A variant of Dorling cartograms are Demers cartograms which use squares instead of circles. Demers cartograms also do not maintain correct adjacencies and disturb relative positions even more than Dorling cartograms. We concentrate on a fourth type of cartograms, *rectangular cartograms*, as introduced by Raisz in 1934 [15], where each region is represented by a rectangle and adjacencies are maintained as well as possible.

Quality Criteria. Whether a rectangular cartogram is good is determined by several factors. One of these is the *cartographic error* [5], which is defined for each region as $|A_c - A_s| / A_s$, where A_c is the area of the region in the cartogram and A_s is the specified area of that region, given by the geographic variable to be shown. Another factor are the *correct adjacencies* of the regions of the cartogram. This requires that the dual graph of the cartogram is the same as the dual graph of the original map. Here the *dual graph* of a map—also referred to as *adjacency graph*—is the graph that has one node per region and connects two regions if they are adjacent, where two regions are considered to be adjacent if they share a 1-dimensional part of their boundaries (see Fig. 3). A third factor is important for the recognizability of a rectangular cartogram: the *relative position* of the rectangles. For example, a rectangle representing the Netherlands should lie west of a rectangle representing Germany. To measure how well a cartogram matches the spatial relations between regions in the input map we use the *bounding box separation distance* (BBSD) [2], which is defined in the next section. Finally, it is important that the *aspect ratio* of the rectangles does not exceed a certain maximum since otherwise the areas become difficult to judge.

Rectangular Duals. We follow the general approach set out in previous work [2,16,18] and construct rectangular cartograms by first finding a suitable *rectangular dual* of the dual graph of the input map. A rectangular dual is defined as follows. A *rectangular partition* of a rectangle R is a partition of R into a set \mathcal{R} of non-overlapping rectangles such that no four rectangles in \mathcal{R} meet at one common point. A *rectangular dual* of a plane graph G is a rectangular partition \mathcal{R} , such that (i) there is a one-to-one correspondence between the rectangles in \mathcal{R} and the nodes in G ; (ii) two rectangles in \mathcal{R} share a common boundary if and only if the corresponding nodes in G are connected.

Not every plane graph has a rectangular dual. A plane graph G has a rectangular dual \mathcal{R} with four rectangles on the boundary of \mathcal{R} if G is an *irreducible*



Fig. 2. Two rectangular duals of the dual graph of a map of Europe (from [2])

triangulation: (i) G is triangulated and the exterior face is a quadrangle; (ii) G has no separating triangles (a 3-cycle with vertices both inside and outside the cycle) [11][2]. A plane triangulated graph G has a rectangular dual if and only if we can augment G with four external vertices such that the augmented graph is an irreducible triangulation.

The dual graph F of a typical geographic map can be easily turned into an irreducible triangulation in a preprocessing step. F is in most cases already triangulated. We triangulate any remaining non-triangular faces (for example the face formed by the nodes for Colorado, Utah, New Mexico, and Arizona). It remains to preprocess internal nodes of degree less than four, such as Luxembourg, Moldova, or Lesotho. In these cases, we add the region to one of its neighbors.

A rectangular dual is not necessarily unique. Consider the two rectangular duals of the dual graph G of a map of Europe shown in Fig. 2. To ensure that G is an irreducible triangulation, Luxembourg and Moldova have been removed. Furthermore, “sea regions” have been added to improve the shape of the outline. The left dual will lead to a recognizable cartogram, whereas the right dual (with France east of Germany and Hungary north of Austria) is useless as basis for a cartogram. Most irreducible triangulations have in fact exponentially many different rectangular duals which are described by *regular edge labelings*.

Regular Edge Labelings. The equivalence classes of the rectangular duals of an irreducible triangulation G correspond one-to-one to the *regular edge labelings* (RELs) of G . An REL of an irreducible triangulation G is a partition of the interior edges of G into two subsets of red and blue directed edges such that: (i) around each inner vertex in clockwise order we have four contiguous sets of incoming blue edges, outgoing red edges, outgoing blue edges, and incoming red edges; (ii) the left exterior vertex has only blue outgoing edges, the top exterior vertex has only red incoming edges, the right exterior vertex has only blue incoming edges, and the bottom exterior vertex has only red outgoing edges (see Fig. 3, red edges are dashed). Kant and He [11] show how to find a regular edge labeling and construct the corresponding rectangular dual in linear time.

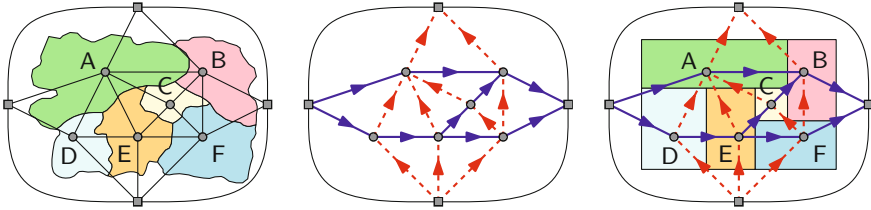
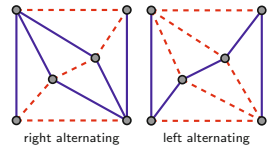


Fig. 3. A subdivision and its augmented dual graph G , a regular edge labeling of G , and a corresponding rectangular dual (from [2])

An *alternating 4-cycle* is an undirected 4-cycle in which the colors of the edges alternate between red and blue. There are two kinds of alternating 4-cycles, depending on the color of the interior edges incident to the cycle. If these are the same color as the next clockwise cycle edge the cycle is *right alternating*, otherwise it is *left alternating*. Fusy [7] proved that the set of RELs of a fixed irreducible triangulation form a distributive lattice. The flip operation consists of switching the edge colors inside a right alternating 4-cycle, turning it into a left alternating 4-cycle. An REL with no right alternating 4-cycle is called *minimal*; it is at the bottom of the distributive lattice.



Although an irreducible triangulation can have exponentially many RELs and hence exponentially many rectangular duals this does not imply that an error free cartogram for this graph exists. The area specification for every rectangle, as well as other criteria for good cartograms, may make it impossible to realize. The lattice structure of the RELs allows us to traverse the space of all RELs for a given graph and find the best rectangular dual for a given set of input values to be realized. However, already for small graphs it is unfeasible to test all possible rectangular duals: the dual graphs of the countries of Europe or the contiguous states of the US both have over four billion labelings. This calls for search strategies that efficiently explore a significant part of the lattice structure. In this paper we present a new search algorithm based on evolution strategies which clearly outperforms previous approaches.

Related Work. The only algorithm for standard cartograms that can be adapted to handle rectangular cartograms is Tobler’s pseudo-cartogram algorithm [17] combined with a rectangular dual algorithm. However, Tobler’s method is known to produce a large cartographic error and is mostly used as a preprocessing step for cartogram construction [13]. The first method for the automated construction of rectangular cartograms was presented by Van Kreveld and Speckmann [18]. Their cartograms have small cartographic error but require (mildly) disturbed adjacencies to realize most data sets. Their method searches through a comparatively small user-specified subset of the RELs. Every labeling in this subset is considered acceptable with respect to the relative positions of the countries. Speckmann *et al.* [16] improved on their earlier results by using an iterative linear programming method to build a cartogram from an REL. With this

methodology world maps could be realized, although small disturbances in the adjacencies were still necessary to obtain acceptable cartographic errors. Speckmann *et al.* [16] used the same user-specified subset of the RELs as Van Kreveld and Speckmann [18]. In a recent paper [2] we presented the first method which uses a heuristic search strategy, namely simulated annealing, on the complete lattice of RELs. We restricted ourselves solely to cartograms with correct adjacencies and nevertheless improved upon the cartographic error of the resulting maps.

A different approach was taken by Inoue *et al.* [10] who compute rectangular and rectilinear cartograms by triangulating the regions and transforming the triangles to meet the desired area requirements. Their rectilinear cartograms have high region complexity and their rectangular cartograms exhibit large cartographic errors. Finally, Heilmann *et al.* [9] gave an algorithm that always produces regions with the correct areas; but the adjacencies can be disturbed badly.

Results and Organization. In this paper we show how to employ evolution strategies to search effectively in the exponentially large lattice of RELs. We find rectangular duals that allow us to realize rectangular cartograms with correct adjacencies and (close to) zero cartographic error. This is a considerable improvement over previous methods. In Section 2 we describe our evolution strategies and in Section 3 we present and discuss an extensive set of experiments.

2 Evolution Strategies

The dual graph of a map can have an exponential number of valid RELs, hence we turn to meta-heuristics to find good solutions in this huge search space. In this section, we present a new approach based on evolution strategies that performs significantly better than our previous method based on simulated annealing [2].

Evolution strategies are an optimization technique that is heavily inspired by natural selection. They use a population of candidate solutions, from which the next generation is constructed by selecting promising individuals and mutating these. If the population is initialized with random solutions, this leads to a broad initial search that quickly focuses on promising regions of the search space. The individuals for our problem consist of valid RELs of the augmented dual graph of our input map. The validity requirement is important, as it reduces the search space by an exponential factor. The population is initialized with semi-random individuals, by starting at the minimum labeling and flipping $d \left(\frac{1}{2} + \frac{r}{8} \right)$ random left alternating 4-cycles, where d is the diameter of the lattice and r is a standard normal distributed random number. Since the lattice of RELs is distributive, every upward path between the same two RELs has the same length and therefore the diameter is simply the number of left alternating 4-cycles we need to flip until we reach the maximum labeling from the minimum labeling. We compute the minimum labeling using a linear-time algorithm by Fusy [6].

After this initialization, every generation follows the same three steps:

1. Compute the fitness scores of all individuals. If the quality measure gives a higher score to better individuals, use this score directly, otherwise (as is

the case with cartographic error and bounding box separation distance) use $1/m$, where m is the score given by the measure.

2. Copy the best 4% of the current population to the next generation. This ensures that the best solutions stay in the population unmodified.
3. Fill the remainder of the next generation by repeating the following process:
 - Use rank selection to select an individual from the current population. The individuals are sorted by fitness in decreasing order. Each individual is assigned a score of 0.9^i , where i is the individual’s rank, so the best individual gets a score of 0.9 and so on. Then each individual is selected with probability equal to the proportion of their score to the total score. Since the selection depends only on the rank of the individuals and not on the fitness values themselves, it is a good choice for optimization using user-specified fitness measures.
 - With probability 0.05, generate a standard normal distributed random number r . If r is positive, move $\frac{dr}{6}$ steps up the lattice, by flipping random left alternating 4-cycles. If r is negative, move $\frac{dr}{6}$ steps down the lattice, by flipping random right alternating 4-cycles. This is a drastic mutation that is used to keep the population from stagnating too much.
 - With probability 0.9, flip a random alternating 4-cycle. This is a small mutation, used for local exploration of the neighbourhood of the selected individual.

Finally, the best REL found during the process is returned. The parameter values presented can be slightly changed to increase performance on various maps and quality measures, but the presented values were found to work well for our instances.

Quality Measures. We now explain how we capture the quality criteria for rectangular cartograms in our evolution strategy. To create a cartogram from an REL we follow the iterative linear programming method presented in [16] with correct adjacencies. Since we consider only valid RELs of the dual graphs of our input maps, this implies that all cartograms we generate have correct adjacencies. That is, all regions that share borders on the geographic input map will do so in the cartogram and regions that do not share borders will not be adjacent in the cartogram. Furthermore, we bound the aspect ratio of all rectangles by 12.

To make a rectangular cartogram as recognizable as possible, it is important that the directions of adjacency between the rectangles of the cartogram follow the spatial relation of the regions of the geographical map as closely as possible. Since these directions of adjacency are specified by the REL, we can assess the recognizability of a rectangular cartogram by looking at its REL. We use the *bounding box separation distance* (BBSD) [2] to quantify how well the directions of adjacency match the geographical directions. The BBSD measures the distance the bounding boxes of the regions would need to be moved to separate them in the direction indicated by the edge label (see Fig. 4).

Finally, to compute the fitness score of an individual we used the weighted sum of 0.7 times the average of squared cartographic errors and 0.3 times the average of squared bounding box separation distances of its regions.

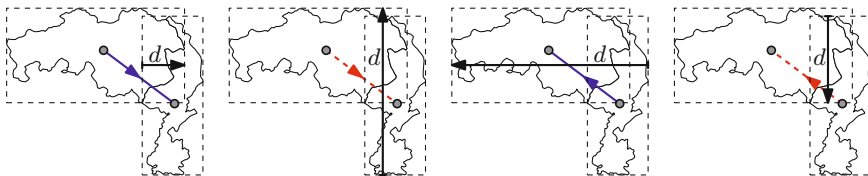


Fig. 4. The BBSD measures the distance d which the bounding boxes of the regions need to be moved to separate them in the direction indicated by the edge label (arrow)

3 Experimental Results

We evaluated our method on a large variety of data sets. For each data set, we measured the cartographic error, bounding box separation distance and running time. We generated cartograms based on three different geographical maps: the contiguous states of the US, the countries of Europe and the countries of the world with a population over 1 million. For the US we used data from the US Census Bureau State and County quickfacts¹. Since cartograms can not easily represent negative or zero values, we used all 45 data sets from the 2010 census where each state was assigned a positive value. Additionally we used the results of the US presidential election of 2008². For Europe we used data from the ranked CIA World Fact Book data sets³. We used all 19 ranked WFB data sets that have data for all countries of Europe included in our cartograms⁴. Our final cartogram uses the world population data from Worldmapper⁵. We conclude with a direct comparison with our previous method².

We generated 20 cartograms for each data set. For each run we recorded the average cartographic error, the maximum cartographic error, and the bounding box separation distance. We summarized these results by taking the average, minimum and maximum over all runs per data set in Table 11. For the US census data we included only the population and geography data sets in the summary, the other data sets show similar trends. The columns ‘min’ give the average cartographic error, maximum cartographic error and the bounding box separation distance of the best cartograms generated for the data set. Since we need only one cartogram per data set, we focus on the values in the ‘min’ columns.

The rectangular cartograms in the figures have regions that are colored based on their error. Shades of red show that a region is too small and shades of blue

¹ <http://quickfacts.census.gov/qfd/index.html>, accessed 2011/11/22.

² <http://elections.nytimes.com/2008/results/president/votes.html>, accessed 2012/02/06.

³ <https://www.cia.gov/library/publications/the-world-factbook/index.html>, accessed 2011/12/10.

⁴ For the area cartogram we use the area of Russia within Europe, http://en.wikipedia.org/wiki/European_Russia, accessed 2012/02/06.

⁵ http://www.worldmapper.org/data/nomap/2_worldmapper_data.xls, accessed 2012/02/01.

Table 1. Average cartographic error (ACE), maximum cartographic error (MCE) and average squared bounding box separation distance (BBSD) for 2010 US census data (people + geography) and World Factbook data of Europe. Average (avg), minimum (min) and maximum (max) taken over 20 runs of our algorithm.

data set description	ACE			MCE			BBSD		
	avg	min	max	avg	min	max	avg	min	max
<i>US census data 2010</i>									
Resident total population	0.04	0.01	0.08	0.29	0.02	0.76	0.05	0.03	0.10
Resident population (RP) 2000	0.05	0.01	0.14	0.34	0.06	0.74	0.05	0.03	0.10
RP < 5 years, percentage (%)	0.04	0.00	0.09	0.16	0.02	0.34	0.04	0.02	0.06
RP < 18 years, %	0.03	0.02	0.07	0.18	0.06	0.24	0.05	0.02	0.13
RP ≥ 65 years, %	0.04	0.01	0.08	0.18	0.04	0.43	0.05	0.02	0.10
RP: total females, %	0.03	0.00	0.05	0.16	0.00	0.32	0.04	0.02	0.07
RP: White alone, %	0.04	0.00	0.08	0.17	0.02	0.40	0.05	0.03	0.07
RP: Black alone, %	0.06	0.01	0.13	0.36	0.05	0.70	0.05	0.03	0.09
RP: Amer. Indian + Alaska Na., %	0.07	0.02	0.14	0.44	0.21	0.89	0.04	0.03	0.07
RP: Asian alone, %	0.05	0.00	0.11	0.32	0.02	0.73	0.05	0.02	0.09
RP: Two or more races, %	0.03	0.00	0.06	0.15	0.00	0.31	0.06	0.03	0.09
RP: Hispanic or Latino Origin, %	0.05	0.00	0.10	0.27	0.02	0.82	0.05	0.03	0.09
RP: Not Hisp., White alone, %	0.04	0.00	0.08	0.19	0.00	0.49	0.05	0.03	0.13
Same househ. 1 yr ago, % '05-'09	0.04	0.00	0.06	0.16	0.00	0.28	0.04	0.02	0.08
Pl. of birth, foreign born, % '05-'09	0.06	0.01	0.10	0.31	0.04	0.77	0.05	0.03	0.09
Pop. ≥ 5 yrs, % lang. other '05-'09	0.04	0.00	0.08	0.22	0.00	0.55	0.05	0.03	0.08
≥ 25 yrs % high sch. grad. '05-'09	0.03	0.00	0.06	0.15	0.00	0.25	0.04	0.02	0.10
≥ 25 yrs % bachelor's deg. '05-'09	0.04	0.00	0.08	0.21	0.00	0.48	0.05	0.02	0.12
Veterans - total '05-'09	0.03	0.00	0.08	0.14	0.01	0.45	0.04	0.03	0.09
Land area in square miles	0.00	0.00	0.01	0.01	0.00	0.04	0.02	0.02	0.04
Population per square mile	0.08	0.02	0.14	0.65	0.11	1.00	0.05	0.03	0.12
<i>World Factbook: Europe (Dec. 2011)</i>									
GDP (purchasing power parity)	0.00	0.00	0.01	0.01	0.00	0.07	0.08	0.08	0.09
GDP real growth rate	0.11	0.09	0.14	0.60	0.50	1.00	0.09	0.08	0.10
GDP - per capita (PPP)	0.07	0.03	0.10	0.34	0.08	0.67	0.09	0.07	0.11
Electricity - production	0.00	0.00	0.01	0.03	0.00	0.11	0.08	0.07	0.10
Electricity - consumption	0.00	0.00	0.01	0.02	0.00	0.15	0.08	0.07	0.10
Airports	0.00	0.00	0.01	0.02	0.00	0.08	0.08	0.07	0.10
Exports	0.01	0.00	0.04	0.03	0.00	0.25	0.08	0.08	0.09
Roadways	0.01	0.00	0.02	0.03	0.00	0.11	0.09	0.08	0.09
Imports	0.00	0.00	0.02	0.01	0.00	0.08	0.09	0.08	0.10
Inflation rate (consumer prices)	0.05	0.01	0.11	0.30	0.09	0.54	0.09	0.08	0.11
Labor force	0.00	0.00	0.01	0.01	0.00	0.04	0.08	0.08	0.10
Population	0.00	0.00	0.01	0.01	0.00	0.11	0.09	0.08	0.10
Unemployment rate	0.10	0.03	0.18	0.54	0.16	0.94	0.09	0.07	0.10
Area	0.00	0.00	0.01	0.02	0.00	0.05	0.08	0.07	0.09
Telephones - main lines in use	0.00	0.00	0.00	0.01	0.00	0.03	0.08	0.08	0.10
Telephones - mobile cellular	0.00	0.00	0.01	0.01	0.00	0.08	0.08	0.07	0.11
Distr. of family income - Gini Ind.	0.03	0.00	0.05	0.22	0.00	0.65	0.09	0.07	0.09
Current account balance	0.05	0.03	0.11	0.49	0.20	0.81	0.09	0.08	0.10
Commercial bank prime lend. rate	0.05	0.02	0.07	0.29	0.10	0.43	0.09	0.08	0.10

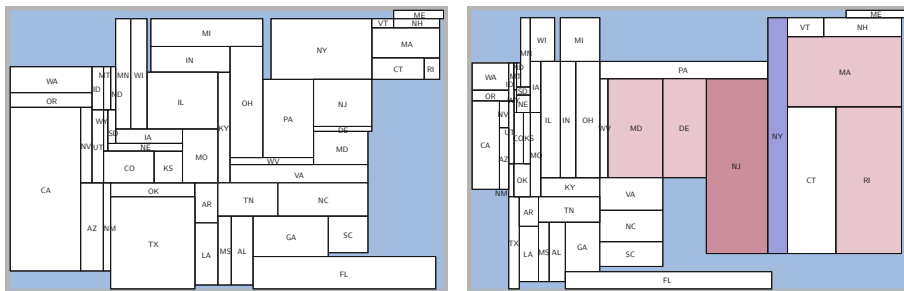


Fig. 5. US population (left) and US population per square mile (right)

show that a region is too large. If the error is below 0.05, the region is white. Note that only Fig. 5 (right) has non-white regions.

All code was written in Java and executed single-threaded, using the OpenJDK Runtime Environment IcedTea6 1.9.9, corresponding to java version 1.6.0_20. For solving linear programs we used IBM ILOG CPLEX 12.0. The measurements of the running time were performed on a 64-bit quad core 1.86GHz Intel Xeon E5320 server with 8 GB RAM, running Ubuntu 10.04.3. On average it took 476 seconds to generate a cartogram for the US, 354 seconds for Europe and 207 minutes for the world. Since the running times showed little variation between data sets, we do not discuss them further.

For all data sets from the US census in the table our algorithm generated at least one map with average cartographic error (ACE) of 2% or less. The average ACE over all runs of the algorithm is between 0% and 8%, where *land area in square miles* has the lowest average and *population per square mile* the highest. For all except two data sets (percentage of American Indian and Alaska Native population and population per square mile) our algorithm generated maps with a maximum cartographic error (MCE) of at most 6% (and an average over all runs of at most 36%). The bounding box separation distance (BBSD) does not vary much and the average over all runs was between 0.02 and 0.06 depending on the data set. For the data sets not included in the table the results are similar, except that there is one data set with a minimum ACE of 4% (wholesale trade) and 4 data sets with a minimum MCE above 7% (Hispanic-owned firms, manufacturing, wholesale trade, and accommodation and food services).

Our rectangular cartogram of the US population in Fig. 5 (left) has an ACE of 0.5%, a MCE of 2.2%, and a BBSD of 0.365. Our results considerably improve on previous work: Van Kreveld and Speckmann [18] obtained a cartogram with an ACE of 8.6% and a MCE of 87.3%, Buchin et al. [2] one with an ACE of 10.2% and a MCE of 59.7%. Inoue et al. [10] don't report on these errors specifically but obtain a rectangular cartogram in which 22 states have a cartographic error between 5% and 20%, and 7 states have a cartographic error larger than 20%.

The data set on population per square mile is one of the few data sets where the MCE obtained is still high (above 7%). Our cartogram in Figure 5 (right)

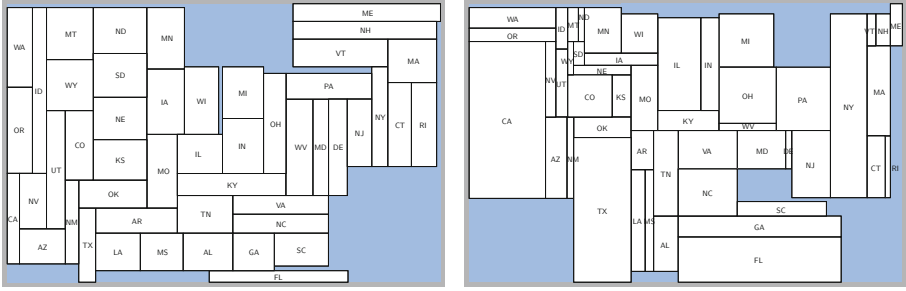


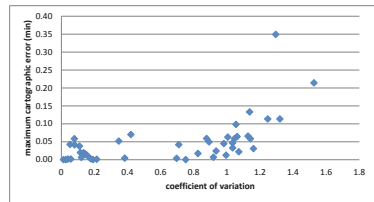
Fig. 6. Percentage of non-Hispanic, white population (left) and number of businesses without payed employees (right). In the left cartogram the correlation to land area is negative, while in the right cartogram the coefficient of variation is high.

has an ACE of 2%, a MCE of 11.3%, and a BBSD of 0.376. In the cartogram we see several causes for the comparably high MCE. In terms of the global layout, the northwest requires so much space (relative to its actual size) that little room is left for the remaining states. The northwest still has not enough space, while the remaining states are depicted with fairly narrow rectangles. More locally, the largest problems seem to be around Pennsylvania, which has to accommodate 4 neighbors with very high population density (and 2 neighbors with lower population density).

In the following we analyze the causes for high MCE further. In terms of the global layout, population density bears several challenges: it is negatively correlated to land area and has a large variation.

Typically cartograms for land area can be generated easily because regions use nearly the same area as on a regular map. It seems natural that data which is negatively correlated to land area is difficult to depict in a cartogram. In our results, however, there does not seem to be a such a relation. Fig. 6 (left) shows a typical cartogram for which land area and the variable depicted have a high negative correlation. The variable is the percentage of non-Hispanic, white population. The cartogram has 0% ACE and MCE, and a BBSD of 0.381.

Generally, high variation in a variable does not necessarily make a variable difficult to depict in a rectangular cartogram. Land area has high variation but can typically be depicted well. Our experiments, however, do indicate a relation between variation and high cartographic error. The scatterplot on the right shows the coefficient of variation



(standard deviation divided by mean) for the data sets from the US census plotted against the best MCE error achieved. While the MCE does not seem to change for coefficients up to about 1, beyond that the maximum cartographic error increases considerably. The population density has a coefficient of 1.3. Another data set with a high coefficient is nonemployer businesses (typically

self-employed individuals). The coefficient of variation for this data set is 1.2. For this data set we did obtain a cartogram shown in Fig. 6 (right) with low cartographic error. Here the ACE is 0.7%, the MCE 3.1% and the BBSD 0.371.

Our final cartogram of the US is a rectangular cartogram showing the results of the US presidential election of 2008. The area of each state corresponds to the number of electoral votes. States won by the Republicans are depicted in red, while states won by the Democrats are depicted in blue. Note that Nebraska does not have a winner-takes-all system, and therefore is two-colored.

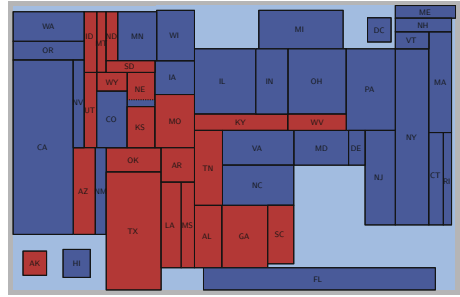


Fig. 7. The US electoral college 2008

We next turn to the data sets for Europe. To ensure that the dual graph of the map is an irreducible triangulation we joined Luxembourg and Belgium, and Moldova and Ukraine. For most data sets we obtained cartograms without cartographic error, see, for example, the population cartogram on page 1. For 6 data sets, however, the MCE was relatively high, namely between 8% and 50%. This is caused by either unproportionately high or unproportionately low values for the countries in the southeast.

For the maps with very low cartographic error, there is still variation in terms of the layout. Fig. 8 shows two cartograms for European exports. The cartogram on the left-hand side has no cartographic error and a BBSD of 0.088. The cartogram on the right-hand side has ACE 0.2%, MCE 1.6% and a BBSD of 0.078. It seems that it is easier to recognize Europe in the cartogram on the right. Hence this cartogram might be preferable despite a small cartographic error.

Our final cartogram shows the world population in 2002. Fig. 9 compares the rectangular cartogram generated by our method to a non-rectangular cartogram from the Worldmapper collection. Our cartogram has ACE 1.17% and MCE 18.5%. Note that we also use a lower percentage of sea area. Overall,

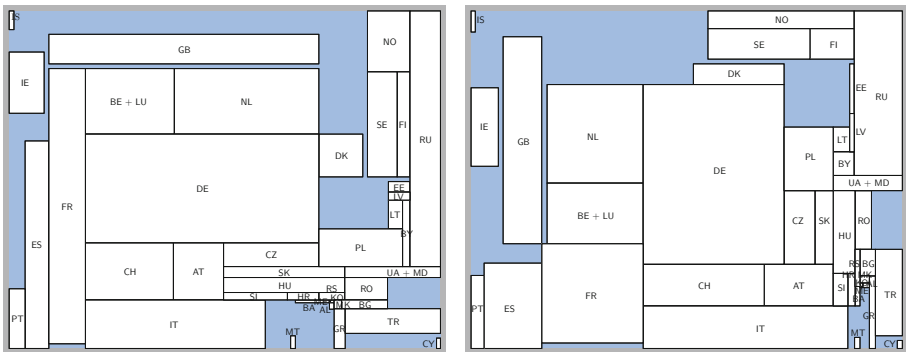


Fig. 8. EU exports: no cartographic error (left) and low cartographic error (right)

Table 2. Evolution strategy vs. simulated annealing approaches. The values are average (avg), minimum (min) and maximum (max) of the average squared bounding box separation distances of the world over 100 runs.

Algorithm	avg	min	max
Simulated annealing	0.101	0.064	0.117
Bootstrapped simulated annealing	0.041	0.019	0.096
Evolution strategy	0.017	0.013	0.025

recognizability is high for this cartogram, with the most noticeable distortion being the abnormal orientation of Russia. This is a change we noticed in all low-error world population maps. It is unlikely that these orientations would have been considered for a hand-picked set of directions, which demonstrates the clear advantage of searching the entire lattice.

We now compare our previous simulated annealing approach [2] to our new evolution strategy. Both use a probabilistic walk over the lattice of regular edge labelings (RELS) to find good solutions, using the fact that neighbouring labelings are likely to be similar in quality. The largest difference is that the evolution strategy starts many random walks simultaneously and concentrates on the promising ones, while simulated annealing performs a single guided walk.

Results of a comparison are given in Table 2. The goal was to find a REL of the world with a low average squared bounding box separation distance. Simulated annealing was run for 10000 steps, while the evolution strategy was given a population size of 50 with 200 generations, resulting in the same number of fitness evaluations. The original simulated annealing was started at the minimum labeling each time. We also include a bootstrapped version of the simulated annealing approach in the comparison that starts at a random labeling. This random labeling was chosen in the same way as labelings in the initial population of the evolution strategy. The evolution strategy significantly outperforms both simulated annealing versions. Not only is the best REL it finds better than the best RELs found by the simulated annealing versions, its average quality is even better than the best quality found by the others. This is caused mainly by improved reliability, which can be seen from the far smaller range of qualities. The evolution strategy has only a 0.012 difference between the best and worst REL, compared to 0.053 and 0.077 for the simulated annealing versions.

4 Conclusion

We presented a new method based on evolution strategies for generating rectangular cartograms with correct adjacencies. The resulting cartograms—for a large range of data sets for Europe, US, and the world—have (close to) zero cartographic error and high visual quality. This is a considerable improvement over previous methods. Nevertheless, several challenges remain. Data sets with

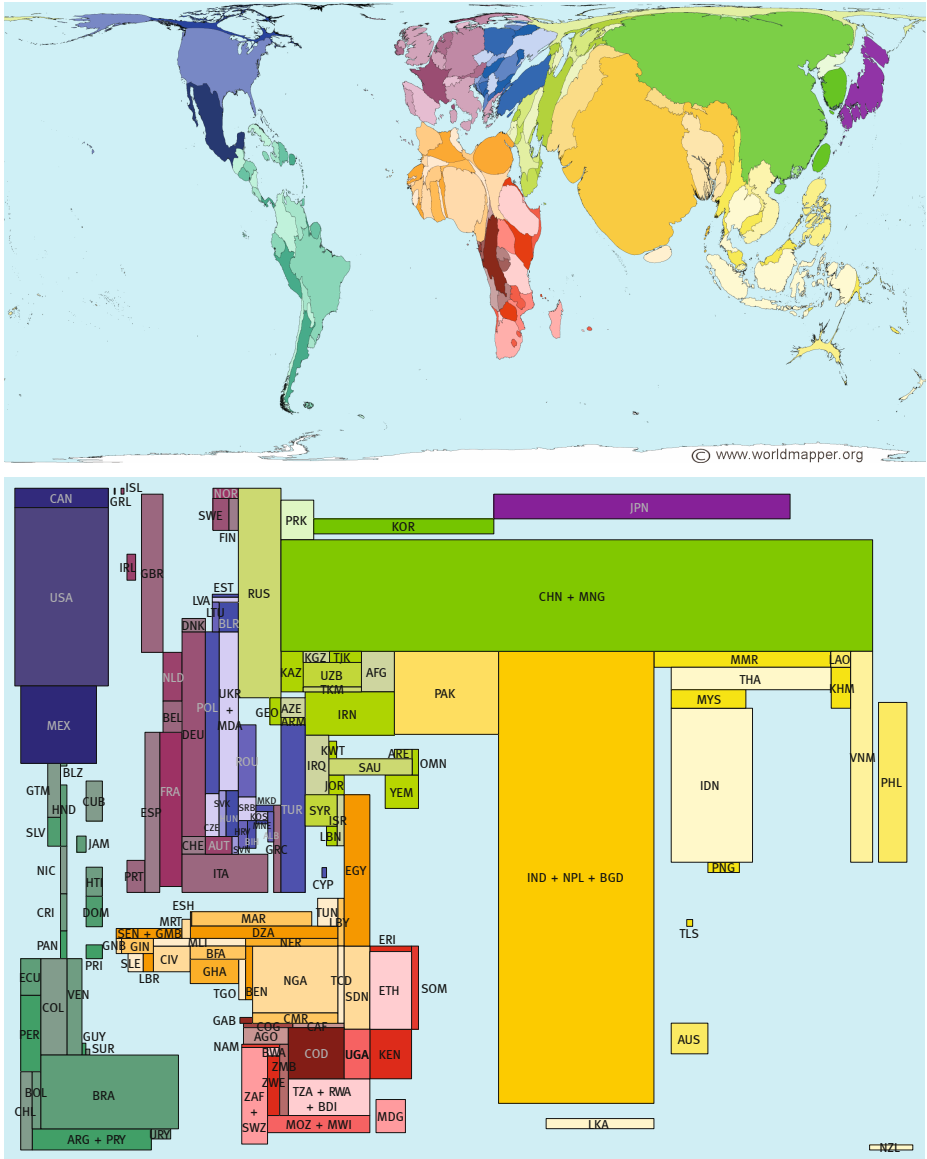


Fig. 9. World population 2002. Top: © Copyright 2006 SASI Group (University of Sheffield) and Mark Newman (University of Michigan)

extremely high variability still prove difficult to realize as cartograms with correct adjacencies, low error, and reasonable aspect ratio. Generally speaking, we would like to be able to search in the lattice of RELs for cartograms with

the best visual properties. These require different trade-offs between adjacencies, relative positions, aspect ratio and error for every data set and it is a challenge to automatically adapt the fitness function to the requirements of each input.

References

1. Bhasker, J., Sahni, S.: A linear algorithm to check for the existence of a rectangular dual of a planar triangulated graph. *Networks* 7, 307–317 (1987)
2. Buchin, K., Speckmann, B., Verdonchot, S.: Optimizing Regular Edge Labelings. In: Brandes, U., Cornelsen, S. (eds.) *GD 2010. LNCS*, vol. 6502, pp. 117–128. Springer, Heidelberg (2011)
3. Dent, B.D.: *Cartography - thematic map design*, 5th edn. McGraw-Hill (1999)
4. Dorling, D.: *Area Cartograms: their Use and Creation. Concepts and Techniques in Modern Geography*, vol. 59. University of East Anglia, Environmental Publications, Norwich (1996)
5. Dougenik, J.A., Chrisman, N.R., Niemeyer, D.R.: An algorithm to construct continuous area cartograms. *Professional Geographer* 3, 75–81 (1985)
6. Fusy, É.: *Combinatoire des cartes planaires et applications algorithmiques*. PhD thesis, École Polytechnique (2007)
7. Fusy, É.: Transversal structures on triangulations: A combinatorial study and straight-line drawings. *Disc. Math.* 309(7), 1870–1894 (2009)
8. Gastner, M., Newman, M.: Diffusion-based method for producing density-equalizing maps. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)* 101(20), 7499–7504 (2004)
9. Heilmann, R., Keim, D.A., Panse, C., Sips, M.: Recmap: Rectangular map approximations. In: *Proceedings of the IEEE Symposium on Information Visualization (INFOVIS)*, pp. 33–40 (2004)
10. Inoue, R., Kitaura, K., Shimizu, E.: New solution for construction of rectilinear area cartogram. In: *Proceedings of 24th International Cartography Conference. CD-ROM* (2009)
11. Kant, G., He, X.: Regular edge labeling of 4-connected plane graphs and its applications in graph drawing problems. *Theoretical Computer Science* 172(1-2), 175–193 (1997)
12. Koźmiński, K., Kinnen, E.: Rectangular dual of planar graphs. *Networks* 5, 145–157 (1985)
13. NCGIA/USGS. *Cartogram Central* (2002), http://www.ncgia.ucsb.edu/projects/Cartogram_Central/index.html
14. Olson, J.: Noncontiguous area cartograms. *Professional Geographer* 28, 371–380 (1976)
15. Raisz, E.: The rectangular statistical cartogram. *Geographical Review* 24, 292–296 (1934)
16. Speckmann, B., van Kreveld, M., Florisson, S.: A linear programming approach to rectangular cartograms. In: *Progress in Spatial Data Handling: Proc. 12th International Symposium on Spatial Data Handling*, pp. 529–546. Springer (2006)
17. Tobler, W.: Pseudo-cartograms. *The American Cartographer* 13, 43–50 (1986)
18. van Kreveld, M., Speckmann, B.: On rectangular cartograms. *Computational Geometry: Theory and Applications* 37(3), 175–187 (2007)

Context-Aware Similarity of Trajectories^{*}

Maike Buchin¹, Somayeh Dodge², and Bettina Speckmann¹

¹ Dept. of Mathematics and Computer Science, TU Eindhoven, The Netherlands
m.e.buchin@tue.nl, speckman@win.tue.nl

² Dept. of Civil, Environmental, and Geodetic Engineering, Ohio State University
dodge.66@osu.edu

Abstract. The movement of animals, people, and vehicles is embedded in a geographic context. This context influences the movement. Most analysis algorithms for trajectories have so far ignored context, which severely limits their applicability. In this paper we present a model for geographic context that allows us to integrate context into the analysis of movement data. Based on this model we develop simple but efficient context-aware similarity measures. We validate our approach by applying these measures to hurricane trajectories.

Keywords: Movement data, geographic context, similarity measures.

1 Introduction

Over the past years the availability of devices that can be used to track moving objects (e.g., GPS systems) has increased dramatically, leading to an explosive growth in movement data. Objects being tracked range from animals (e.g., for behavioral studies) and cars (for traffic prediction), to hurricanes, sports players, and suspected terrorists. Tracking an object gives rise to a sequence of points in time and space, called a *trajectory*. Naturally the goal is not only to track objects but also to extract information from the resulting data. Consequently recent years have seen a significant increase in the number of methods developed to extract knowledge from moving object data [13].

The movement of animals, people, and vehicles is embedded in a geographic context. This context both enables and limits movement. For instance, cars are constrained to move on road networks and turtles ride ocean currents. Hurricanes cannot develop over cold ocean current and wolves cannot cross a wide river gorge. Contextual information, such as terrain, land cover, street networks, or weather data, is often available. It is crucial to take this context into account when performing movement analysis. Consider trajectories of migratory birds collected over weeks. A clustering algorithm might detect and reject certain

^{*} M. Buchin and B. Speckmann are supported by the Netherlands Organisation for Scientific Research (NWO) under project no. 612.001.106 and no. 639.022.707, respectively. S. Dodge was supported in parts by Forschungskredit University of Zurich (Credit No. 57060804), and NASA grant number NNX11AP61G.

trajectories as outliers with malfunctioning sensors, when in fact a storm forced a group of birds to deviate from the usual path.

A fundamental analysis task on trajectories is similarity analysis. It answers the question: “How similar are the movements paths of two or more objects?” Similarity analysis can be the basis of other tasks, such as clustering, pattern recognition, simplification, or representation. Also, it can be an analysis task by itself, as for instance in hurricane analysis (see below and Section 4).

Our goal in integrating context into the analysis are two-fold. We want to learn about the movement from the context (e.g., an animal heading towards a goal made a detour because of an obstacle), and we want to learn about the context from the movement (if all tracks avoid an area, there is likely an obstacle there). We develop context-aware similarity measures for the first task: understanding movement based on context. These similarity measures allow to distinguish trajectories by their spatial component as well as their context.

Contribution. Our contribution is two-fold. In Section 2, we present a model for geographic context that allows us to integrate context into the analysis of movement data. In Section 3, we then develop simple but efficient context-aware algorithms for similarity analysis based on this model. The context of two trajectories clearly plays a significant role in similarity analysis. However, to the best of our knowledge, we present *the first context-aware approach to trajectory similarity for movement not constrained to networks*.

To validate our approach we apply our context-aware similarity measures to hurricane tracks (see Section 4). Hurricanes are known to follow similar paths. Therefore, when a new hurricane evolves, meteorologists use past hurricanes with a similar initial track for predicting the track of the developing hurricane. Hurricanes are also known to be strongly influenced by geographic context, most importantly land/sea, geographic latitude, surface temperature, and surface pressure [9]. Hurricanes whose tracks are very similar in shape, may still be very different in their nature. Consider for example the hurricanes in Figure 1. Spatially, their trajectories are very similar. However, they differ in geographic context (land/sea). When a hurricane hits land, its energy source – the warm sea surface – is taken away, which will severely weaken it. Thus, it is crucial to distinguish these hurricanes, as our context-aware similarity measures do.

Related Work. Most algorithms for analyzing trajectories have so far ignored context: trajectories are analyzed in an empty space [14]. This has been identified as one of the pitfalls of current methods for movement analysis [12]. An exception is the analysis of trajectories on road networks and subway systems. Here the known underlying network reduces the dimensionality of the problem and leads to more efficient algorithms and more meaningful results. However, for movement not constrained to a network, hardly any context-aware analysis algorithms exist. A notable exception is the work by Andrienko *et al.* [2] which uses an event based model as opposed to the geometric model we propose here.

There are many context-free approaches to measure the similarity of trajectories, e.g. [8] and references therein. Besides geographic context, also temporal context can influence movement, e.g., people sleeping at night, and birds

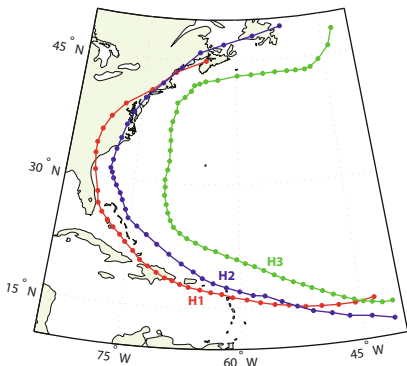


Fig. 1. Trajectories of hurricanes distinguished by context: tropical storm Chris 1988 (red), hurricane Bertha 1996 (blue), and Edouard 1996 (green)

migrating in Spring. Contrary to geographic context, time is a natural component of trajectories, and some similarity measures take time into account [6,10,15,18].

2 Modeling Context

In this section we present a model for geographic context that allows us to integrate context into the algorithmic analysis of movement data. We first describe the various types of geographic context relevant for movement data together with suitable models for each type of context.

Network. Some entities are constrained to move on a network, e.g., cars on roads, trains on tracks, boats on rivers, whereas other entities may be constrained to cross a network only at certain points, e.g., people on foot.

Model: labeled geometric graph.

Land cover. The type of land cover influences for instance the speed of an object, e.g., a hurricane is faster on water than on land.

Model: labeled polygonal subdivision.

Obstacles. Some parts of geographic space are impassable for some entities.

Model: set of polygons.

Terrain. The slope and altitude of a location influence movement, e.g., cyclists are faster downhill than uphill.

Model: grid or tin.

Ambient Attributes. Geographic or meteorological attributes, such as temperature and humidity.

Model: point, grid, or vector data.

Other agents. Presence of other agents can cause the emergence of certain movement patterns (e.g. attraction and competition among animals lead to particular behavioral patterns such as courtship or fighting, respectively).

While other agents definitely influence movement, we will not discuss them further in this paper, since they are not a form of geographic context.

Obstacles may be part of a network or land cover. Obstacles and attributes can also be modeled as labeled polygonal subdivisions. For this, obstacles are modeled as a subdivision of obstacles and non-obstacles. Attributes are modeled as a subdivision into zones of equal attribute values. Also, several types of geographic context (e.g., landcover, properties of the terrain, attributes) can be treated as further attributes of a trajectory. That is, each point of the trajectory can be annotated by the geographic context value, e.g., type of landcover, slope, temperature. However, this will not reveal if two points are in the same zone of attribute values, i.e., the same region of landcover, or slope, or temperature.

Context may be *discrete* or *continuous*, i.e., it takes on discrete values, such as landcover, or continuous values, such as temperature. This distinction plays a role when comparing different contexts. However, trajectories are typically discrete themselves. Furthermore, context may be *dynamic* or *static*, i.e., it may change over time or not. A changing context is important to take into account when comparing trajectories that occurred at different times.

Context influences movement in different ways. We distinguish:

1. whether context *limits* or *enables* movement
2. and whether it does so *fully* or *partially*

A bridge over a gorge enables movement, whereas obstacles restrict movement. Sometimes context has a full impact on movement: a bridge may be absolutely necessary to cross a gorge. In other cases, context has a partial impact on movement: birds flying with air currents, or steep slopes in the terrain.

Our model is based on the movement paradigm by Nathan *et al.* [16]. A moving entity has an internal state (why move?), a navigation capacity (when and where to move?) and a motion capacity (how to move?), and it is influenced by external factors (the environmental context). These four components interact to produce the movement path. For instance, an animal may be motivated by thirst (internal state) to move to a waterhole (external factor). Here, the waterhole enables the movement of the animal, whereas further properties of geographic space, such as obstacles, may limit the movement. Depending on the physical state of the animal, a waterhole may be fully or partially enabling: the animal needs to drink to stay alive, or it would also survive without drinking at this waterhole. Similarly, a road network may limit movement *fully* or *partially*: a car will always stay on it, whereas a tractor may leave it to go on a field. The distinction between *full* or *partial* influence of context is essential: when we know a context has full influence, we can for instance use it to detect outliers.

Although the movement paradigm by Nathan *et al.* is originally introduced for Movement Ecology, we believe that a similar paradigm also applies to other domains. In particular, this is the case for hurricane movements. A hurricane only forms under favorable climatic conditions in terms of wind speed, air pressure, and sea surface temperature. It cannot form on land, or on cold water. Usually, when a hurricane hits land, its movement direction and speed changes. Therefore, the internal state, navigation and motion capacities of hurricanes are highly related to external factors (ambient attributes and geographic context).

3 Context-Aware Similarity Measure

Our goal is to define a similarity measure that takes into account the geographic context in the comparison of movement paths. For this, we first ask the question: ‘How does geographic context influence the similarity of trajectories?’ We claim that a fundamental influence of geographic context is that it may distinguish trajectories. For example, context distinguishes the hurricanes in Figure 1. The most basic situations that can occur are shown in Figure 2: in (a) two entities are moving in areas of different context, e.g., one on water, the other on land. In (b) two entities may be moving in areas of the same type of context (e.g., land), however, they are separated by a region of different context (e.g., a river). These trajectories may appear similar when the geographic context is *not* taken into account, but they differ when the geographic context is taken into account.

We will consider geographic context here that is modeled as a labeled polygonal subdivision. As discussed in Section 2, this may model land cover, obstacles, or attributes aggregated to zones. Thus it is an important model covering many types of geographic context. Not covered by this model are networks, terrains, and attributes. Nevertheless, this model can be adapted for both terrain and attributes using classification (e.g., topographic contour lines and attribute classes). For network-constrained data, approaches to determine trajectory similarity exist [11, 19]. For attributes a straightforward, alternative approach is a multi-dimensional approach, however this may not be suitable in all cases.

Note that geographic context has further implications on similarity than distinguishing trajectories with different contexts. Geographic context may influence attributes of a movement paths (such as speed or sinuosity). For instance, a person typically walks slower on sand than on asphalt. Thus, two spatially close trajectories on sand/asphalt may not be considered similar when using a speed-dependent similarity measure. Here, we address similarity measures that distinguish trajectories with differing contexts. We see this as the most fundamental influence of geographic context on trajectory similarity.

Problem statement. We are given two trajectories, and a labeled polygonal subdivision of the area in which the trajectories move. We want to define similarity measures for trajectories that take into account the geographic context modeled by the labeled polygonal subdivision.

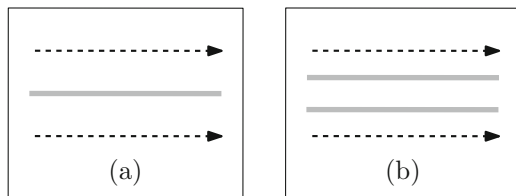


Fig. 2. Trajectories (in dashed, black) over a geographic context (in bold, gray)

3.1 Approaches

A trajectory in our setting has a spatio-temporal part (its position in time and space) as well as a contextual part (its position in the labeled polygonal subdivision). Generally, we see three approaches to context-aware movement similarity analysis: these two parts can be treated as

1. *equal* and similarity computed in multi-dimensional space,
2. *independent* and similarity computed separately,
3. *integrated* and similarity computed in an integrated way.

Next, we briefly discuss the first two approaches and compare all three approaches. We conclude that an integrated approach is most suitable and give a solution for this in Section 3.

For the equal approach, the context parts (position in a labeled polygonal subdivision) are mapped to numerical values for a (possibly weighted) multi-dimensional analysis. Note that typically no straight-forward such mapping will exist. Here, the mapping (and possibly weighing) determines the relative weight of context vs. space and time.

For the independent approach, the trajectory is split into two: a (context-free) spatio-temporal trajectory, and a (pure) context trajectory. The context trajectory would be the sequence of labeled cells of the subdivision that the trajectory visits (and corresponding time stamps). For example, consider trajectory A: $\{(x_1, y_1, t_1, C_1), (x_2, y_2, t_2, C_1), (x_3, y_3, t_3, C_3)\}$. Its spatio-temporal part is $\{(x_{a1}, y_{a1}, t_{a1}), (x_{a2}, y_{a2}, t_{a2}), (x_{a3}, y_{a3}, t_{a3})\}$ and its context part is $\{C_1, C_1, C_3\}$, respectively. Known similarity measures can be applied to the spatio-temporal trajectory and the context trajectory separately. This gives two distance values: a spatial distance and a context distance. These can then be combined using an additive (weighted sum) or multiplicative (weighted average) approach, or one distance can be used as filter for the other.

Comparison of approaches. We claim that the equal approach is not appropriate for two reasons. First, mapping context to numerical values loses information. Second, space, time, and context are not equal. The independent approach may be applicable in some cases, however, we claim that some cases require an integrated approach. Consider, for instance, the (abstract) situation in Figure 3. Four trajectories A, B, C, D are shown over a subdivision of two cells. Trajectories A, B are closest spatially, but differ in context. Trajectories C, D are close with respect to context, but differ spatially. Trajectories B, C differ, when considering context and space separately. However, when considering space and context jointly, trajectories B, C are the most similar. Trajectories B, C are first close spatially, but separated by context, then they are close in context, but with a larger spatial distance. Imagine for instance that these are monkeys running through forest and grass. First, they run along the edge of a forest (on different sides), then they both run over an open grass field (at a larger distance). In the following Section 3, we develop similarity measures that capture this similarity.

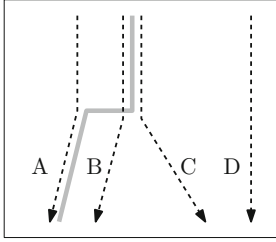


Fig. 3. Trajectories (in dashed, black) over a geographic context (in bold, gray)

3.2 Integrated Similarity Measures

Now we show how to extend existing similarity measures to make them context-aware, by integrating context and spatial distance. The main idea is to define the distance between two points as their spatial distance plus their context distance. Intuitively, this means it “costs” to cross context boundaries. Points with equal contexts, i.e., in the same cell of the subdivision, will get zero context cost. Thus, for equal context the distance equals the spatial distance.

Note that adding costs makes sense only if the costs have comparable scales. Thus, we require to be able to compare spatial and context distance. If these are incomparable, then an integrative approach, which outputs one distance value, seems infeasible. For convenience, we introduce a scaling parameter for the context distance, which we call the *context scale*. This allows us to first define a context distance, and then relate it to the spatial distance by setting the context scale. The value of the context scale will depend on the application. We discuss this with an example of hurricanes in more detail in Section 4.

Based on this notion of integrated point-to-point distance, we propose a framework for context-aware trajectory similarity consisting of three ingredients

1. a spatial distance, e.g., Euclidean distance
2. a context distance, (see below)
3. a distance measure based on point-to-point distances, e.g., Fréchet distance

Choosing all of these ingredients results in a context-aware similarity measure for trajectories. That is, our approach extends known distance measures (3) to make them context-aware, by adding a spatial distance (1) and a context distance (2). If all three ingredients are metrics, so is the resulting measure. In this approach, we take into account that space, time and context are not equal. We use time in the overall distance measure to determine the matching of points. The relative weight of space and context is determined by the scaling parameter.

Next, we first discuss different options for a context distance. Then, we discuss how to compute the resulting context-aware similarity measures. As distance measure we consider three popular measures for trajectory similarity that are based on spatial point-to-point distances: comparing distances at equal times [6,10,15,18], the Fréchet distance [3,4,5], and the Hausdorff distance [17].

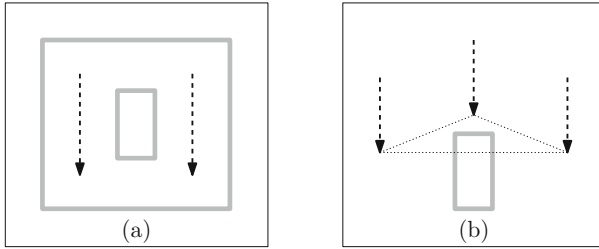


Fig. 4. Trajectories (in dashed, black) over a geographic context (in bold, gray)

Context Distance. We propose to use a cost between cells of the subdivision as context distance. That is, the context distance of two points is the distance of the cells they lie in. An alternative would be to use a context distance between points. The disadvantage of a distance between cells is that this will ignore “islands”, see Figure 4(a). The two trajectories, though separated by a cell of the subdivision, still lie in the same cell. Thus, their context distance will be zero. A context distance between points could, for instance, consider the context along (shortest) paths between the points and thus detect the island.

The advantage of a distance between cells is that the resulting distance measure is a metric. This property is not necessarily maintained by a context distance between points, see Figure 4(b). Suppose as context distance between two points we add a cost for each subdivision boundary crossed by a shortest path between the points. Then, as Figure 4(b) shows, the context distance does not fulfill the triangle inequality (the distance from A to C via B is less than the distance from A to C). To remedy this, one could use *geodesic shortest paths*, i.e., allow paths to go around islands. This may lead to “jumping” over islands, that is, islands may increase the spatial distance, not the context distance.

Summarizing, we propose a distance between cells, as it gives an intuitive and sound definition. In particular, it handles the cases in Figure 2 and Figure 3.

Choices. We propose four different context distances between cells. These are based on two independent choices:

labels unit cost or cost dependent on label of the cell,

paths unit cost or cost of shortest path between cells.

The first choice refers to whether we assign a cost depending on the labels of the subdivision. A unit cost means the cost between cells do not depend on the label. Alternatively, the costs may depend on the label. For instance, imagine the subdivision models land cover. Then we may choose to give a higher cost between *grass* and *water* than between *grass* and *wood*. If we choose a cost dependent on the labels, we still want to maintain the triangle inequality. That is, we choose costs $c(L_1, L_2)$ between labels L_1, L_2 such that for all three labels L_1, L_2, L_3 holds $c(L_1, L_3) \leq c(L_1, L_2) + c(L_2, L_3)$. A unit cost would assign an equal cost to all different labels (and zero to equal labels). This makes sense, when the relation between labels is not known.

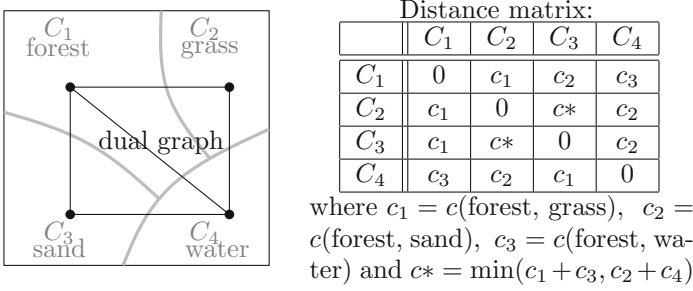


Fig. 5. Example of context distance along shortest paths

The second choice refers to whether we assign the same costs between any two cells of the subdivision (possibly depending on the labels), or whether we assign the cost dependent on the shortest path between the cells. For the second choice, we consider the *dual graph* of the subdivision. That is, we consider the graph G , where each cell C constitutes a vertex of the graph, and edges exists between neighboring cells. A shortest path then refers to a path of minimal cost, where the cost of each edge is determined by the first choice, that is, either a unit cost, or a cost depending on the labels of the cells. For an example see Figure 5.

Computation. The proposed context-aware similarity measures can be computed by extending algorithms for Hausdorff, Fréchet distance and equal time distance in three ways

1. computing the context distance matrix (if using path lengths between cells),
2. locating points in the subdivision and (if necessary) refining trajectories,
3. adding context costs when computing the distance measure.

The first two steps are pre-processing to the main algorithm in step 3.

Computing the context distance matrix. This is known as the *all-pairs shortest paths* problem. The fastest known algorithms for this problem on planar graphs run in (sub)quadratic time. However, in practice, slower, but simpler algorithms, e.g., the Floyd-Warshall algorithm with a cubic runtime may be preferred. In particular, this holds if the size of the subdivision is (much) smaller than the size of the trajectories, and the algorithm in step 3 dominates the runtime.

Locating points in the subdivision and refining trajectories. If we do not need to refine trajectories, then we only need to compute in which cells the vertices of the trajectories lie. For this we can use a standard point location data structure like a trapezoidal map. Computing this data structure takes $O(m \log m)$ preprocessing time. A point location query, that is reporting the cell of a given trajectory point, then takes $O(\log m)$ time for a subdivision of size m , with $O(m)$ space requirements. Thus, this takes $O(n \log m)$ time for a trajectory of size n .

To refine trajectories, we also need to find all intersections of trajectories with subdivision boundaries. For this, we use known algorithms to preprocess the subdivision for *ray shooting queries*: given a point of which we know the location

and a ray starting at that point, we want to know where this ray intersects the subdivision. We can do the preprocessing step in $O(m)$ time, where m is the size of the subdivision. Queries then take $O(\log m)$ time. We can locate the first trajectory vertex in $O(\log m)$ time using a point location data structure. Then we find the intersections of the first trajectory edge and the location of the second trajectory vertex using a ray shooting query from the first vertex in direction of the second vertex. If we intersect a cell boundary before reaching the second point, we continue from there. After reaching the second vertex, we process the remaining trajectory in the same way. The running time of this is $O(h \log m)$ per shoot for h intersections. Thus, we need $O((n + h) \log m)$ time in total.

In practice, we expect that trajectories do not often intersect the subdivision. In this case, also simpler strategies apply, for example as described in Section 4.

Adding context costs when computing the distance measure. Algorithms for Hausdorff and equal time distance are straightforward to extend by simply adding the context cost to the spatial cost. For the Fréchet distance, the decision algorithm based on the *free space diagram* [1] can be extended as follows. Trajectories need to be segmented at context boundaries, as described above. Then each trajectory edge lies completely in one cell. Thus, each free space cell (corresponding to two trajectory edges) gets a constant extra context distance. We simply add this in each free space cell. The runtime of the algorithm remains $O(n^2 \log n)$. Note that refining trajectories may increase their complexity. However, in practice we typically expect not more than a linear number of intersections, thus not affecting the asymptotic runtime. For computing the Fréchet distance, a set of critical values is searched, employing the decision algorithm in each step [1]. Critical values are distances between points on the trajectories. Here, we again simply add the context distance.

Fréchet Distance in Weighted Regions. The Fréchet distance has been extended for paths in weighted regions [7]. There the cost of a path is the weighted sum of path lengths in each weighted region. Our model of adding context costs when crossing context boundaries can be “simulated” by this model, as follows: give each context boundary a width ϵ (for a small $\epsilon > 0$) and weight $(c_i + 1)$. Give each cell the weight 1. Then a path of length ℓ crossing b boundaries has weight $\ell + \sum_{i=1}^b c_i$ in both models. Thus, the algorithms from [7] can solve our problem. However, these algorithms give approximative solutions and have much higher running times (more than $O(n^4)$).

4 Evaluation: Test on Hurricane Data

In the previous section, we proposed context-aware similarity measures for trajectories, which are extensions to known measures. We implemented this extension for the Fréchet distance, and tested it on hurricane tracking data. In this section we present our experimental results.

4.1 Context of Hurricanes

For hurricanes, similarity is an interesting analysis task, which is for instance relevant for predicting hurricane paths (see Section 4). Hurricanes are known to be influenced by geographic context, which can be distinguished as follows:

external factors: temperature, barometric pressure, land/sea, topography

internal factors: intensification, wind speed, move speed, diameter

We tested our measure on land/sea as geographic context. However, our method applies to any context that can be modeled as labeled polygonal subdivision. This includes continuous values, such as temperature, by aggregation to zones.

4.2 Description of Data

We tested our method on hurricane tracking data over land/sea. We considered hurricanes in the North Atlantic Basin in the years 1995, 2004, and 2005. The data was obtained from NOAA National Hurricane Center¹. The hurricanes are tracked every 6 hours (00:00, 06:00, 12:00, 18:00). The chosen years had predominant hurricane activities with 17 storms in 1995, 11 storms in 2004, and 20 storms in 2005, thus 48 in total. Furthermore, we used a geographic data set containing the coast lines for the polygonal subdivision into land/sea.

Preprocessing. We cut the hurricanes at Longitude 55° W at start and end, to ensure that entire hurricanes locate in a similar spatial region (see Fig. 6a). Large differences in starts and ends would otherwise dominate the distance. Next, we located and annotated trajectory points in the subdivision and computed intersection points of trajectories with the coastlines. For this, since we have a sparse subdivision, we first split the coast line into constant size pieces. Then we build an R-Tree of bounding boxes of these pieces, and query in this structure.

4.3 Similarity Measure

We chose the Fréchet distance as distance measure since it compares the shape of trajectories well. As context distance we chose a unit distance between different labels (only option between two labels (land/sea)) and a shortest path distance between cells (with paths of length at most two, given only one large sea cell). We varied the value of the context scale (see below). Thus, in terms of Section 3, we used the ingredients: (1) Euclidean distance, (2) shortest path distance with unit costs between different labels, and different context scales, (3) Fréchet distance.

Context Scale. Recall that the context scale is used to weigh the context distance, thus putting the spatial and context distance in relation. In particular a context scale of zero implies ignoring context. One can interpret context scale as follows: Two hurricanes with spatial distance close to zero but differing context are considered as similar as two hurricanes with equal context and a spatial distance of the value of the context scale. Here we used context scales 0, 300, and 500 km, which we chose based on the hurricanes spatial distances (see Table 4).

¹ www.nhc.noaa.gov/

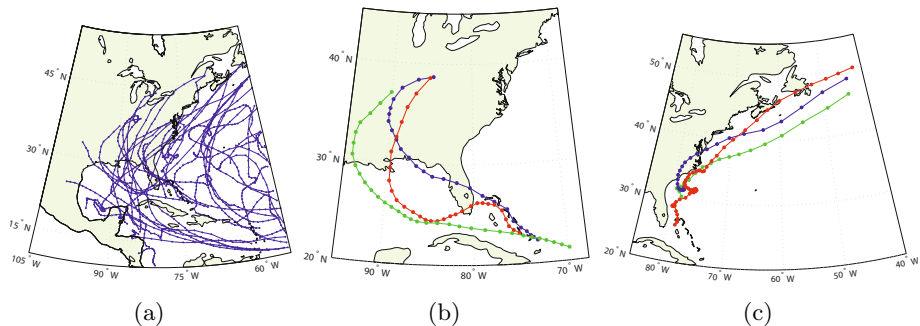
Table 1. Computed distances and ranks of the eight spatially closest pairs of hurricanes for three different context scales. (‘-’ denotes rank > 10).

H1	H2	context	rank (0)	rank (300)	rank (500)	distance (0)	distance (300)	distance (500)
Harvey	Nate	water	1	1	1	248km	248km	248km
Alex	Gaston	water/land	2	4	-	315km	561km	761km
Chantal	Luis	water	3	2	2	489km	489km	489km
Erin	Katrina	water/land	4	5	-	482km	614km	814km
Alex	Ophelia	water/land	5	-	-	531km	808km	1008km
Erin	Rita	water/land	6	-	-	550km	850km	1050km
Chantal	Irene	water	7	3	3	551km	551km	551km
Katrina	Rita	water/land	8	-	-	572km	698km	898km

4.4 Experimental Evaluation

We computed the context-aware distances of three years of hurricane data (1995, 2004, and 2005), using the similarity measure described above. Table 1 shows the results for three different context scales (0km, 300km, 500km) of the eight spatially closest pairs of hurricanes. Besides the distance the table also gives the rank in the order by distance. That is, rank k means that a pair of hurricanes is the k^{th} in the order by distances for this context scale. Thus, the rank gives an indication of relative distances. For instance, Chantal – Luis and Chantal – Irene move higher up in the order (have smaller rank) for larger context scale. This is because they have equal context (water), and therefore the context-aware distance is not affected by the context scale (see their distance values).

For our analysis, hurricanes crossing land are more interesting. There are two triples of these among the most similar hurricanes: Erin – Rita – Katrina (Fig. 6b) and Alex – Gaston – Ophelia (Fig. 6c). For a high context scale (500), all hurricanes with differing contexts become less similar (ranks > 10). For a moderate context scale (300), two pairs of hurricanes remain among the first 10

**Fig. 6.** (a) Data set. (b) Erin 1995 (blue), Katrina 2005 (red), and Rita 2005 (green). (c) Alex 2004 (green), Gaston 2004 (blue), and Ophelia 2005 (red).

ranks: Alex – Gaston and Erin – Katrina. Alex, Gaston, and Ophelia have very similar paths, with Gaston crossing Carolina and Virginia, and the other two staying on water. More interesting is the triple Erin – Rita – Katrina. We note two effects: (i) Erin – Katrina remain more similar (rank 5 at context scale 300), while Erin – Rita and Katrina – Rita become less similar (rank > 10 at context scale 300); (ii) the order of similarity changes: at context scale 0, Erin – Rita are more similar than Katrina – Rita, and vice versa at context scales 300 and 500. Both of these effects mirror the different development of the hurricanes: (i) Erin and Katrina first cross Florida before re-emerging in the Gulf of Mexico and making secondary landfall. They weaken over land, then re-intensify as they move back over the water. In contrast, Rita does not cross over Florida, making its first landfall between Louisiana and Texas. (ii) Katrina, in contrast to Erin, crosses only the tip of Florida and loops longer of the gulf before making landfall again. Rita has a similar path over the gulf. Thus, our context-aware similarity measure better captures the actual similarity of these hurricanes.

5 Conclusion and Future Work

We proposed context-aware similarity measures for trajectories, which extend known similarity measures. We tested our methods on hurricane tracking data and conclude that our method is fast, simple, and effective. That is, it distinguishes hurricanes that are spatially close but not close in their context.

We see several paths for future work. We plan to apply our ideas on integrating context into movement analysis to more knowledge discovery tasks. We also intend to assess the applicability of our method on other types of movement data, such as animal movements. Furthermore, context-aware similarity measures that do not “ignore islands” are an interesting open question, which we plan to investigate. Finally, the aspect of robustness to small changes in context has not been addressed yet.

Acknowledgements. The authors are grateful to Jane Strachan for her insights on hurricane similarity, the anonymous referees for helpful comments, and the European COST Action IC0903 MOVE for supporting the short term scientific mission of S. Dodge.

References

1. Alt, H., Godau, M.: Computing the Fréchet distance between two polygonal curves. *International Journal of Computational Geometry and Applications* 5, 75–91 (1995)
2. Andrienko, G., Andrienko, N., Heurich, M.: An event-based conceptual model for context-aware movement analysis. *International Journal of Geographical Information Science* 25, 1347–1370 (2011)
3. Brakatsoulas, S., Pfoser, D., Salas, R., Wenk, C.: On map-matching vehicle tracking data. In: *Proc. 31st International Conference on Very Large Data Bases*, pp. 853–864 (2005)

4. Buchin, K., Buchin, M., Gudmundsson, J.: Constrained free space diagrams: a tool for trajectory analysis. *International Journal of Geographical Information Science* 24, 1101–1125 (2010)
5. Buchin, K., Buchin, M., Gudmundsson, J., Löffler, M., Luo, J.: Detecting commuting patterns by clustering subtrajectories. *International Journal of Computational Geometry and Applications* 21(3), 253–282 (2011)
6. Buchin, K., Buchin, M., van Kreveld, M.J., Luo, J.: Finding long and similar parts of trajectories. *Computational Geometry: Theory and Applications* 44(9), 465–476 (2011)
7. Cheung, Y.K., Daescu, O.: Fréchet Distance Problems in Weighted Regions. In: Dong, Y., Du, D.-Z., Ibarra, O. (eds.) *ISAAC 2009*. LNCS, vol. 5878, pp. 97–111. Springer, Heidelberg (2009)
8. Dodge, S.: Exploring Movement Using Similarity Analysis. PhD thesis, University of Zurich (2011)
9. Elsner, J., Kara, A.: *Hurricanes of the North Atlantic: Climate and society*. Oxford University Press (1999)
10. Frentzos, E., Gratsias, K., Theodoridis, Y.: Index-based most similar trajectory search. In: *Proc. 23rd IEEE International Conference on Data Engineering*, pp. 816–825 (2007)
11. Hwang, J.-R., Kang, H.-Y., Li, K.-J.: Searching for Similar Trajectories on Road Networks Using Spatio-temporal Similarity. In: Manolopoulos, Y., Pokorný, J., Sellis, T.K. (eds.) *ADBIS 2006*. LNCS, vol. 4152, pp. 282–295. Springer, Heidelberg (2006)
12. Laube, P., Purves, R.: How fast is a cow? Cross-scale analysis of movement data. *Transactions in GIS* 15(3), 401–418 (2011)
13. Miller, H.J., Han, J.: *Geographic Data Mining and Knowledge Discovery*, 2nd edn. Taylor & Francis Group (2009)
14. Mountain, D.: The dimensions of context and its role in mobile information retrieval. *SIGSPATIAL Special* 3, 71–77 (2011)
15. Nanni, M., Pedreschi, D.: Time-focused clustering of trajectories of moving objects. *Journal of Intelligent Information Systems* 27, 267–289 (2006)
16. Nathan, R., Getz, W.M., Revilla, E., Holyoak, M., Kadmon, R., Saltz, D., Smouse, P.E.: A movement ecology paradigm for unifying organismal movement research. *Proc. National Academy of Sciences of the United States of America* 105(49), 19052–19059 (2008)
17. Nutanong, S., Jacox, E.H., Samet, H.: An incremental Hausdorff distance calculation algorithm. In: *Proc. 37th International Conference on Very Large Data Bases*, vol. 4(8), pp. 506–517 (2011)
18. Sinha, G., Mark, D.M.: Measuring similarity between geospatial lifelines in studies of environmental health. *Journal of Geographical Systems* 7(1), 115–136 (2005)
19. Tiakas, E., Papadopoulos, A., Nanopoulos, A., Manolopoulos, Y., Stojanovic, D., Djordjevic-Kajan, S.: Searching for similar trajectories in spatial networks. *Journal of Systems and Software* 82(5), 772–788 (2009)

Generating Named Road Vector Data from Raster Maps

Yao-Yi Chiang¹ and Craig A. Knoblock²

¹ University of Southern California,
Information Sciences Institute and Spatial Sciences Institute
4676 Admiralty Way, Marina del Rey, CA 90292, USA
yaoyichi@isi.edu

² University of Southern California,
Department of Computer Science and Information Sciences Institute
4676 Admiralty Way, Marina del Rey, CA 90292, USA
knoblock@isi.edu

Abstract. Raster maps contain rich road information, such as the topology and names of roads, but this information is “locked” in images and inaccessible in a geographic information system (GIS). Previous approaches for road extraction from raster maps typically handle this problem as raster-to-vector conversion and hence the extracted road vector data are line segments without the knowledge of road names and where a road starts and ends. This paper presents a technique that builds on the results from our previous road vectorization and text recognition work to generate named road vector data from raster maps. This technique first segments road vectorization results using road intersections to determine the lines that represent individual roads in the map. Then the technique exploits spatial relationships between roads and recognized text labels to generate road names for individual road segments. We implemented this approach in our map processing system, called Strabo, and demonstrate that the system generates accurate named road vector data on example maps with 92.83% accuracy.

Keywords: Raster map, road vectorization, text recognition, named road vector data, map labeling.

1 Introduction

Cartographers have been making maps for centuries and road maps are one of the most used maps among all map types. Today we have access to a huge number of map collections in raster format from a variety of sources. For instance, the United States Geological Survey (USGS) has been mapping the United States since 1879. The USGS topographic maps at various time periods cover the entire country and contain informative geographic features, such as contour lines, buildings, and road lines. These raster maps are easily accessible compared to other geospatial data (e.g., road vector data) and present a unique opportunity for obtaining road information for the areas and time periods where and when

road vector data do not otherwise exist. For example, we can generate named road vector data (road vector data that have a road-name attribute) from historical maps and build an accurate geocoder [Goldberg et al., 2009] or a gazetteer for spatiotemporal analysis of human-induced changes in the landscape.

Generating named road vector data from raster maps is challenging for a number of reasons. First, maps typically contain overlapping layers of geographic features, such as roads, contour lines, and text labels. Thus, the map content is usually highly complex and presents a difficult task for converting the road geometry in raster maps to vector format. Second, maps contain characters of various sizes constituting multi-oriented text labels, which cannot be recognized using classic optical character recognition (OCR) techniques. Finally, even after the road geometry is vectorized and text labels are recognized, there still exists the problem of labeling individual road lines with the recognized labels.

This paper presents an approach to generate named road vector data from raster maps while requiring minimal user effort. Figure 1 shows our overall approach, which integrates our previous map processing work (the interactive road vectorization [Chiang and Knoblock, 2011a] and text recognition techniques [Chiang, 2010; Chiang and Knoblock, 2011b]) and offers a new contribution: an automatic technique to identify individual road segments from the road vectorization results and then associate the recognized road labels with the road segments. This technique is the reverse engineering of cartographic-labeling methods [Agarwal et al., 1998; Doddi et al., 1997; Edmondson et al., 1996; Freeman, 2005]. The resulting named road vector data can be used in a geographic information system (GIS).

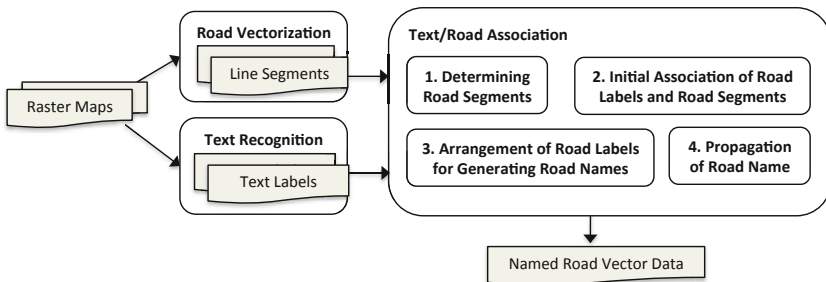


Fig. 1. The overall approach for generating named road vector data from heterogeneous raster maps

The remainder of this paper is organized as follows: Section 2 describes our previous map processing work on which the techniques in this paper built, Section 3 presents this paper's main contribution on associating road vector data with road labels, Section 4 reports on our experimental results, Section 5 discusses the related work, and Section 6 presents the discussion and future work.

2 Previous Work

This section briefly reviews our previous work on text recognition [Chiang, 2010; Chiang and Knoblock, 2011b] and road vectorization [Chiang and Knoblock, 2011a] from raster maps.

2.1 Text Recognition

In our previous work, we developed an interactive text recognition approach that requires only minimal user effort for processing heterogeneous raster maps [Chiang, 2010; Chiang and Knoblock, 2011b]. This approach first exploits a few examples of text areas for extracting text pixels and locating individual text strings. Figure 2 shows our user interface for labeling example text areas. Figure 3 shows an example map and the results where individual text strings are identified and shown in distinct colors (the color is only for explaining the idea). Once individual text strings are identified, we automatically detect the string orientations and rotate the strings to horizontal to then leverage conventional OCR software for recognizing the horizontal strings.



Fig. 2. Our user labeling interface for text recognition from raster maps



Fig. 3. Identify individual text labels

2.2 Road Vectorization

In our previous work, we developed an interactive road vectorization approach that requires minimal user effort to handle heterogeneous raster maps [Chiang and Knoblock, 2011a]. Similar to our text recognition approach, this road vectorization technique exploits a few examples of road areas to extract road pixels and generate road vector data.

To identify the road colors in a raster map for extracting the road pixels, our approach asks a user to first select a few example areas of roads. An example area of a road is a rectangle that is centered at a road intersection or a road segment. We exploit the fact that the road pixels in an example area of roads are a portion of one or more linear objects that are near the area center to determine the colors that represent roads in a map.

With the separated road layer (i.e., the set of extracted road pixels), we automatically detect the road width and format (i.e., single-line or double-line roads) and then dynamically generate parameters for applying the morphological operators (i.e., the dilation, erosion, and thinning operators) to extract and rebuild the road geometry (i.e., the centerline representation of the road network). The left image of Figure 4 shows an example map and the middle image shows the extracted road geometry, where the road lines near the intersections are distorted as a result of applying the morphological operators on thick lines. To extract accurate road vector data around the intersections, we detect the road intersections in the road geometry and label potential distortion areas around the intersections. Finally, we trace the thinned-line pixels outside the distortion areas to reconstruct the road intersections and generate the road vector data. The right image of Figure 4 shows the resulting road vector data where the road geometry around the intersections is accurate.

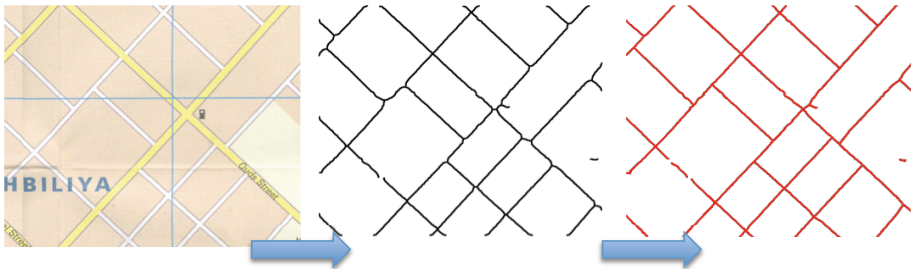


Fig. 4. Extract accurate road geometry and road vector data

3 Association of Road Vector Data and Road Labels

Our text/road association algorithm includes four major components. (i) The first component processes road vectorization results to generate individual road segments. Each road segment contains a set of line segments constituting the same road in a map. (ii) The second component assigns each road label to a road

segment. (iii) For the road segments that are assigned with more than one road label, this component arranges the road labels to generate a road name using the relative positions between the assigned road labels and the road segment. (iv) Finally, the fourth component propagates the road names from road segments that have assigned road labels to the road segments that do not have assigned labels. In addition, if a road name is broken into several parts to label a long road in the input map, the separate parts are merged into a road name.

3.1 Determining Road Segments

The input to our road segmentation algorithm is the road vector data generated from our previous road vectorization work. The extracted road vector data contains a set of line segments, which are short, straight lines, without the knowledge of which line segments belong to each road segment in the map. Since road name changes commonly happen at road intersections, we use the locations of road intersections to group the input line segments into individual road segments – a road segment is a section of a road that is bounded by road endpoints or road intersections where more than two line segments meet. Figure 5 shows an example input and output of our road segmentation algorithm.

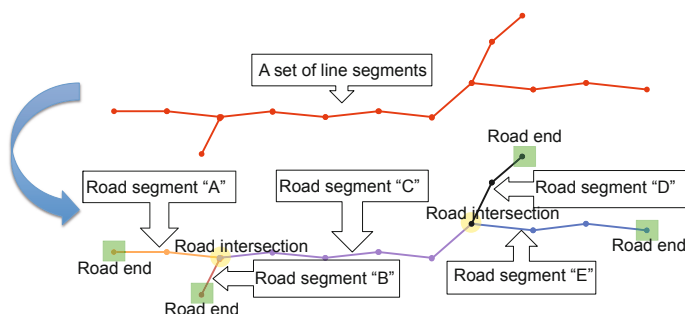


Fig. 5. Determining road segments from line segments based on road intersections

Our road segmentation algorithm first computes the connectivity of the endpoints of every line segments in the input road vector data. If an endpoint connects to only one other endpoint, the endpoint is a road end, namely a *RE* (the green squares in Figure 5). If an endpoint connects to more than two other endpoints, the endpoint is a road intersection, namely a *RI* (the yellow circles in Figure 5). If an endpoint is neither a *RE* nor a *RI*, the endpoint is a connecting point, namely a *CP*.

Once we have the connectivity of every endpoint of the input line segments, we iteratively process every input line segments until every line segment is assigned to a road segment. In one iteration, our algorithm starts from an unprocessed line segment and we first check the connectivity of its two endpoints. If the two endpoints are both classified as either a *RE* or *RI*, we assign the line segment as a road segment itself and then continue to process other line segments. If no

or only one endpoint is classified as either a *RE* or *RI*, we search for the unprocessed line segments that connect to this line segment through the endpoints that are classified as a *CP*. We stop when the connected line segment we found contains a *RE* or *RI* or no line segment exists that is connected to this line segment. Figures 6(a), 6(b), and 6(c) show an example test map, the input road vector data, and the road segmentation results. The red crosses shows the endpoints of the line segments and the endpoints of road segments in Figures 6(b) and 6(c), respectively. The road segmentation results are then used in the next step with the recognized road labels to generate named road vector data. In an



Fig. 6. Example inputs and intermediate results for grouping road segments and determining the locations of road labels

unusual case where the road name changes at non-road intersection locations, user input would be required to further separate the road segments.

3.2 Initial Association of Road Labels and Road Segments

Once we have the road segments, we start to assign each recognized road label to one of the road segments. Figure 6(d) shows the test map where the rectangles show the bounding boxes of the identified road labels. The recognized road labels, together with the identified road segments, are the input to this step.

Map labeling is a well investigated technique in both cartography [Edmondson et al., 1996] and computer science [Agarwal et al., 1998; Doddi et al., 1997; Freeman, 2005]. In general, to label linear features in a map, a computer program or a cartographer places the labels in parallel to the corresponding linear features. The distance between a label and the corresponding feature should be smaller than the distance between the label to any other features of the same kind in the map. Therefore, to determine the correspondence between a road label and a road segment, we first assign every road label to the road segment that is the closest to the label and has the same orientation of the label.

To compute the distance between a road segment and a road label, we use the mass center of the road label to represent the position of this label. We calculate the distance between the mass center to each of the line segments in a road segment and use the shortest line-segment-to-mass-center distance as the distance between the road segment and the road label.

For a road label containing n character pixels, (x_i, y_i) , the road label's mass center, (X_m, Y_m) , is calculated as follows:

$$X_m = \frac{\sum_{i=1}^n x_i}{n}, Y_m = \frac{\sum_{i=1}^n y_i}{n} \quad (1)$$

To determine the parallelism between a road label and a road segment, we compare the orientation of the road segment with the orientation of the road label. The orientation of each road label is determined using our text recognition algorithm [Chiang and Knoblock, 2011b], and we compute the orientation of each road segment as follows: we first utilize the *Least-Squares Fitting* algorithm to find a straight line that best fits each road segment in the two dimensional space and then compute the orientation of the straight line. Assuming a linear function L for a set of points in a road segment, by minimizing the sum of the squares of the vertical offsets between the points and the line L , the *Least-Squares Fitting* algorithm finds the line L that most represents the road segment. For the target line function L as:

$$Y = m \times X + b, \quad (2)$$

the *Least-Squares Fitting* algorithm works as follows:

$$m = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2} \quad (3)$$

and

$$b = \frac{\sum y - m \sum x}{n} \quad (4)$$

With the line function of every road segment, we then derive the orientations of all road segments by applying the inverse trigonometric function, $ArcTan$, to the slope (m) of the road segments' line functions.

Because the orientations of the road labels and road segments are not estimated from the same type of data format (the road labels are a group of pixels and the road segments are vectors), to determine the parallelism between a road label and a road segment, we empirically define a buffer, B , as 10° . If the difference between the orientations of a road label and a road segment is smaller than B , the road label and the road segment is determined to be in parallel.

For curved roads, if the road label is also curved along the curvature of the roads, the estimated orientation of the road label is similar to the road orientation determined using this approach. However, in the case where straight strings are used to label curved roads, the road label would not be assigned correctly and would need manual correction.

3.3 Arrangement of Road Labels for Generating Road Names

We can have multiple road labels assigned to a road segment if a road name is divided into more than one label in the map as shown in Figure 7. In this case, we need to determine the order of the assigned road labels for a road segment to then assemble the ordered road labels for generating a road name.

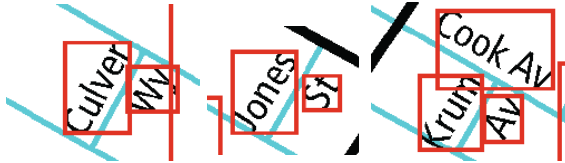


Fig. 7. More than one road labels can be assigned to a road segment

Given a road segment with multiple assigned road labels, for each of the road labels, we first determine which side of the road segment the road label appears in the map. This is determined using the cross product between the two *endpoints*, (X_s, Y_s) and (X_e, Y_e) , of the road segment and the mass center, (X_m, Y_m) , of the road label:

$$((X_e - X_s) \times (Y_m - Y_s)) - ((Y_e - Y_s) \times (X_m - X_s)) \quad (5)$$

The sign of the result from the cross product indicates which side the road label appears in the map. Once we have the relative position between the assigned road labels and the road segment, *we first rotate the road labels to the horizontal direction using the label orientation* and then check the relative position of the road labels and arrange the road labels as follows:

(i) If two road labels appear on the same side of the road segment, we order the labels using their X_m positions. This is because in English writing, a sequence of words is read from the left (a smaller X_m) to the right (a larger X_m).

(ii) If two road labels appear on different sides of the road segment, we order the road labels using their Y_m positions. Similarly, this is because in English writing, a sequence of words should be read from the top (a larger Y_m) toward the bottom (a smaller Y_m). For example, as shown in Figure 7, in the initial assignment, the road labels “Culver” and “Wy” are both assigned to the same road segment. After we rotate both road labels to the horizontal direction, the road label, “Culver”, has a larger Y_m value among the two road labels, so it should be placed in front of “Wy”. This case is also demonstrated using the road names “Jones St” and “Krum Av” in Figure 7.

Once we determine the order of the road labels for a road segment, we concatenate the ordered road labels to generate a merged road name.

3.4 Propagation of Road Names

Generally in computer map labeling and cartography map-making principles, not every road segment in a map is labeled with a road name because of the limited labeling space in the map and to avoid possible overlap of map labels. Therefore, repetitive road names are eliminated to improve the reading experience. For example, the “St. Louis Av” and the “University St” shown in Figure 8 are spread across more than one intersection, but the road names only appear once in the map. The green arrows indicate the possible start and end points of these roads that can be interpreted by a viewer. Moreover, words belong to a road name can be spread out to indicate the extent of a road, such as the “Greer Av” and “Dodier St” in Figure 8.

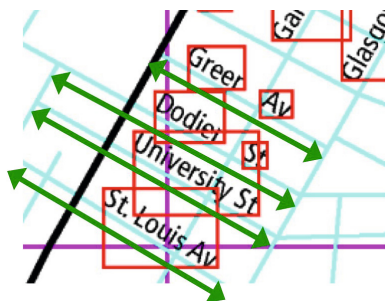


Fig. 8. Road labels do not repeat for each segment

As a result, after every road label is assigned to a road segment and multiple road labels that are assigned to a road segment are merged, we still need to assign road names to the road segments that do not have an assigned road label, and we need to merge the road labels that belong to the same road name but are assigned to more than one road segment. For example, we need to merge

“Greer” and “Av” into “Greer Av” and then assign the merged road name to the corresponding road segments.

We start from a road segment, RS , that has an assigned road label, and we search for any other road segment that is connected to this road segment and has the same orientation. If a connected road segment, $NextRS$, has no assigned road label, we record that $NextRS$ has the same road name as RS . If $NextRS$ has assigned road labels, we order the assigned road labels of RS and $NextRS$ using the described method in the previous section. Then for the ordered road names, A followed by B , if B is a short string, we determine that the combination of A and B represents a road name and hence A and B should be merged. This is because if the last word in the sequence is a short string, this last word very often represents the road-type abbreviations (e.g., Av, St, Pl, Wy, and Dr). We define a short road label as a label of less than 5 characters since the longest abbreviations of the road types are “Blvd”, which are 4 characters. This rule helps to merge two words into a complete road name.

The name propagation algorithm runs iteratively and records the number of road segments that have their road names assigned during each iteration. After an iteration, the algorithm checks if the number of road segments that have their road names assigned has increased. If the number stays the same, the algorithm stops since there are no road names that can be propagated. The results after this name propagation algorithm is a set of road segments, each labeled with a road name or an empty label indicating there is no road label in the map associated with this road segment.

4 Experiments

We have implemented the approach described in this paper in a system called Strabo. This section presents our experiments on Strabo for generating named road vector data from 6 raster maps of 2 map sources. The 2 sources are Rand McNally Maps (RM maps) and Afghanistan Information Management Services (AIMS maps) ¹. Figure 6(a) shows an example RM map. The RM maps are designed for navigation purpose and contain very detailed road information. The RM maps represent common street maps that can be purchased in local gas stations and tourist stores. The AIMS maps contain only the information of major roads and are commonly used in urban planning.

We focus on evaluating the techniques for associating road names to the road vector data. The details of our road vectorization and text recognition results can be found in our previous work [Chiang, 2010; Chiang and Knoblock, 2011a]. To generate named road vector data from RM maps, we labeled 1 road area, 1 text area, and 1 non-text area. For AIMS maps, we labeled 1 road area and 1 text area. Based on these example areas, Strabo converted the road lines in the original maps to vector format, recognized the road labels, and generated named road vector data.

¹ The information for obtaining the test maps and ground truth can be found on:

http://www.isi.edu/integration/data/maps/prj_map_extract_data.html

Strabo recognized 154 road labels and 892 road segments in the RM maps, and 15 road labels and 338 road segments in the AIMS maps. We manually verified each road segment using the test maps. Among the 892 road segments in RM maps, 866 road segments (97.09%) were correctly identified. Among the 338 road segments in AIMS maps, 327 road segments (96.75%) were correctly identified. The incorrect road segments have false road topology and/or geometry. Figure 9 shows an intersection where the road topology was incorrect due to the various road widths of the intersecting road lines. Sharp angles make the intersecting lines closer to each other and hence our road vectorization algorithm could not produce accurate geometry using the morphological operators.

To evaluate the overall performance for generating named road vector data, we define the accuracy as the length of correctly labeled road segments divided by the length of all identified road segments. A correctly labeled road segment is defined as follows: every line segment of a correctly labeled road segment represents a part or all of a road line that has the road name as the assigned name of the road segment. The accuracy for RM maps is 92.38% and for AIMS maps is 93.27%.

Figure 10 shows a portion of the extracted named road vector data displayed and labeled using Esri ArcMap. The yellow lines are the extracted road vector data and the red text with underlines are the assigned names.² From Figure 10, we can see that Strabo successfully propagated the road names to the corresponding road segments so that the road lines that are not labeled in the original map also had correctly assigned road names.

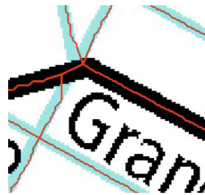


Fig. 9. Examples of incorrectly extracted road topology (red lines are the extracted road lines)

The majority of errors in our experimental results are due to the fact that some extracted road topology and/or geometry are incorrect. During the road vectorization process, the road topology could be incorrectly extracted due to the various road widths of the intersecting road lines. Because of this incorrect road topology, the road names of the connecting roads could not propagate through this intersection and resulted in falsely assigned road names or incomplete road names. Including a manual editing process for the results of the road vectorization and segmentation steps would reduce this type of error.

In addition, OCR could produce recognition errors. For example, in the test map, the string “BLVD” was recognized as “8LVD” and “Parnell St” was

² The map labeling algorithm of ArcMap did not label every road segments.

recognized as “Pamell St”. If one or more characters of a road name was incorrectly recognized, the named road vector data results for the road segments associated with this road name were all considered to be incorrect.

Overall, Strabo generated accurate named road vector data: the average accuracy for the 6 maps from the 2 sources is 92.83%. To improve the results, we could have a user editor to process the extracted road vector data and recognized road labels for quality assurance so the text/road association algorithm could have more accurate input data.

5 Related Work

Map processing is an active area in both academic research and commercial software. However, to the best of our knowledge, the work presented in this paper is one of the first complete approaches to handling the problem of generating named road vector data from raster maps.

The most closely related work is a map computerizing system called MapScan [MapScan, 1998] from the United Nations Statistics Division. MapScan has the functionality for manually converting the linear features in raster maps into vector format and recognizing the text strings in raster maps. MapScan includes an extensive set of image processing tools (e.g., the morphological operators) and labeling functions for the user to manually computerize the raster maps, which requires intensive user input. For example, to recognize text strings using MapScan, the user needs to label the areas of each text string and the string has to be in the horizontal direction. The association between the road names and extracted road vector data is achieved manually. In contrast, our approach requires only minimal user effort for recognizing road labels and extracting road vector data from raster maps, and further, we associate road names to road vector data automatically.

For text recognition from raster maps, [Pouderoux et al., 2007] present a text recognition technique for raster maps. They identify text strings in a map by analyzing the geometric properties of individual connected components in the map and then rotate the identified strings horizontally for OCR. [Roy et al., 2008] detect text lines from multi-oriented, straight or curved strings. Their algorithm handles curved strings by applying a fixed threshold on the connecting angle between the centers of three nearby characters. Their orientation detection method only allows a string to be classified into 1 of the 4 directions. In both [Pouderoux et al., 2007; Roy et al., 2008], their methods do not hold when the string characters have very different heights or widths. Moreover, these approaches handle specific types of road labels and do not work further to determine the association between the recognized road labels and the geographic features in raster maps.

For road vectorization from raster maps, [Bin and Cheong, 1998] extract road vector data from raster maps by identifying the medial lines of parallel road lines and then linking the medial lines. The linking of the medial lines requires various manually specified parameters for generating accurate results, such as

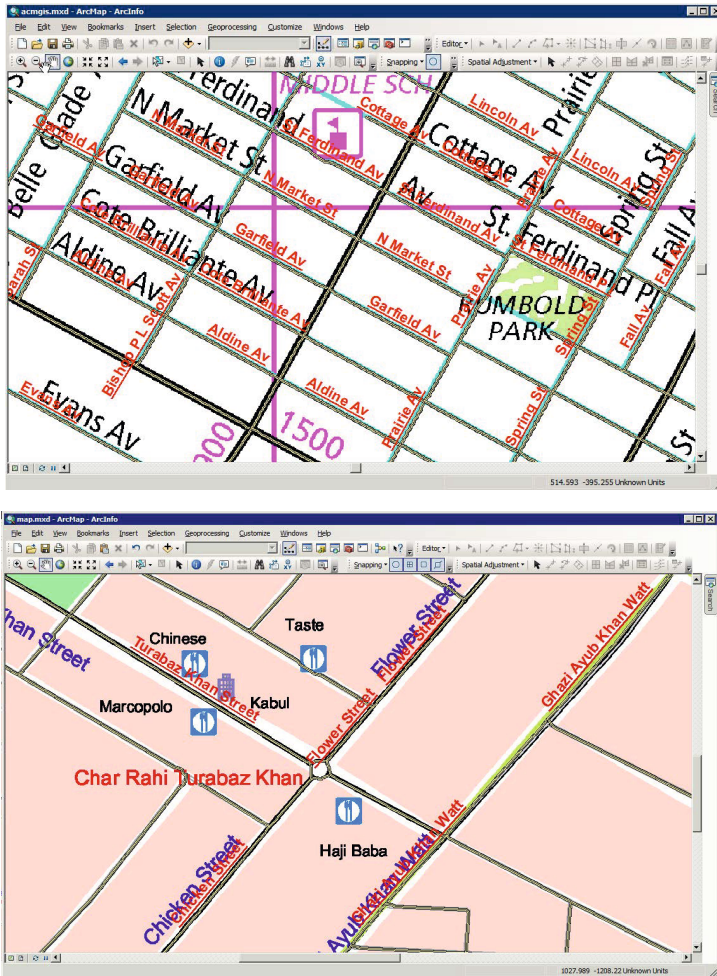


Fig. 10. The resulting named road vector data from RM and AIMS maps

the thresholds to group medial-line segments to produce accurate geometry of road intersections.

[Itonaga et al. \[2003\]](#) focus on non-scanned raster maps that contain only road and background areas. They exploit the geometric properties of roads (e.g., elongated polygons) to first label each map area as either a road or background area. Then they apply the thinning operator to extract a 1-pixel width road network from the identified road areas. The geometry distortions in the thinning results are then corrected by user-specified constraints, such as the maximum deviation between two intersecting lines.

In comparison to the approach in this paper, the techniques of [Bin and Cheong \[1998\]](#) and [Itonaga et al. \[2003\]](#) require significant user effort on parameter

tuning. Moreover, their approaches do not determine where line segments compose a road segment in the vectorization result.

In addition to road vectorization research work, many commercial products offer the functionality for raster-to-vector conversion, such as Vextractor³, Raster-to-Vector⁴ and R2V from Able Software.⁵ However, these commercial products do not have any text recognition capability, and hence do not work for generating named road vector data.

6 Discussion and Future Work

This paper presented a complete approach for generating named road vector data from raster maps. In particular, we presented an approach that automatically identifies individual road segments from road vectorization results and then associates recognized road labels with corresponding road segments. This approach, together with our previous road vectorization and text recognition work, allows a user to use only minimal effort for extracting named road vector data from raster maps. The resulting named road vector data are widely useful, such as for supporting a geocoder, building a gazetteer, and enriching available road information for spatial analysis in a GIS. In the future, we plan to test this work using raster maps with non-English labels. In addition, we plan to exploit the named road vector data generated from historical raster maps for spatiotemporal analysis.

References

- Agarwal, P.K., van Kreveld, M., Suri, S.: Label placement by maximum independent set in rectangles. *Computational Geometry* 11(3-4), 209–218 (1998)
- Bin, D., Cheong, W.K.: A system for automatic extraction of road network from maps. In: *Proceedings of the IEEE International Joint Symposia on Intelligence and Systems*, pp. 359–366 (1998)
- Chiang, Y.-Y.: *Harvesting Geographic Features from Heterogeneous Raster Maps*. PhD thesis, University of Southern California (2010)
- Chiang, Y.-Y., Knoblock, C.A.: A general approach for extracting road vector data from raster maps. *International Journal on Document Analysis and Recognition* (2011a), doi: 10.1007/s10032-011-0177-1
- Chiang, Y.-Y., Knoblock, C.A.: Recognition of multi-oriented, multi-sized, and curved text. In: *Proceedings of the Eleventh International Conference on Document Analysis and Recognition* (2011b)
- Doddi, S., Marathe, M.V., Mirzaian, A., Moret, B.M.E., Zhu, B.: Map labeling and its generalizations. In: *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 148–157 (1997)
- Edmondson, S., Christensen, J., Marks, J., Shieber, S.M.: A general cartographic labelling algorithm. *Cartographica: The International Journal for Geographic Information and Geovisualization* 33(4), 13–24 (1996)

³ <http://www.vextrasoft.com/vextractor.htm>

⁴ <http://www.raster-vector.com/>

⁵ <http://www.ablesw.com/r2v/>

- Freeman, H.: Automated cartographic text placement. *Pattern Recognition Letters* 26, 287–297 (2005)
- Goldberg, D.W., Wilson, J.P., Knoblock, C.A.: Extracting geographic features from the internet to automatically build detailed regional gazetteers. *International Journal of Geographic Information Science* 23(1), 92–128 (2009)
- Itonaga, W., Matsuda, I., Yoneyama, N., Ito, S.: Automatic extraction of road networks from map images. *Electronics and Communications in Japan (Part II: Electronics)* 86(4), 62–72 (2003)
- MapScan: MapScan for Windows Software Package for Automatic Map Data Entry, User's Guide and Reference Manual. Computer Software and Support for Population Activities, INT/96/P74, United Nations Statistics Division, New York, NY 10017, USA (1998)
- Pouderoux, J., Gonzato, J.C., Pereira, A., Guitton, P.: Toponym recognition in scanned color topographic maps. In: *Proceedings of the Ninth International Conference on Document Analysis and Recognition*, vol. 1, pp. 531–535 (2007)
- Roy, P.P., Pal, U., Lladós, J., Kimura, F.: Multi-oriented English text line extraction using background and foreground information. In: *IAPR International Workshop on Document Analysis Systems*, pp. 315–322 (2008)

An Ordering of Convex Topological Relations

Matthew P. Dube and Max J. Egenhofer

School of Computing and Information Science, University of Maine
5711 Boardman Hall, Orono, ME, USA 04469-5711
matthew.dube@umit.maine.edu, max@spatial.maine.edu

Abstract. Topological relativity is a concept of interest in geographic information theory. One way of assessing the importance of topology in spatial reasoning is to analyze commonplace terms from natural language relative to conceptual neighborhood graphs, the alignment structures of choice for topological relations. Sixteen English-language spatial prepositions for region-region relations were analyzed for their corresponding topological relations, each of which was found to represent a *convex* subset within the conceptual neighborhood graph of the region-region relations, giving rise to the construction of the convex ordering of region-region relations. The resulting lattice of the convex subgraphs enables an algorithmic approach to explaining unknown prepositions.

Keywords: Conceptual neighborhood graph, topological spatial reasoning, convex relations, spatial language, spatial prepositions.

1 Introduction

Geographic information science has been advocating a topological understanding of space to assemble cognitively plausible models that mirror the human understanding of spatial phenomena. Within this realm, such models as the 4-intersection [19], the 9-intersection [14], the Region Connection Calculus [39], and conceptual neighborhood graphs [18,24] contribute symbolically to analyzing spatial scenes for similarity and to distinguishing spatial scenes from one another [9,37]. The formally defined spatial relations are mutually exclusive, yielding *atomic* relations (i.e., the smallest currency to describe spatial scenes). They may, however, be combined in *disjunctions* (exclusive ORs) to account for vague scenarios, as often expressed in natural-language terms. While the relations' conceptual neighborhood graphs have been studied substantially, their cognitive plausibility is still an unanswered question. Mathematically based studies [15,18] and initial psychological assessments [29,30] have demonstrated the utility of the graphs, but as of yet, *topological relativity* [28] to bridge formal and observed human spatial cognitive processes has not been determined explicitly.

As a contribution to the discussion about the plausible value of a conceptual neighborhood graph, we assess the structure of a conceptual neighborhood graph for modeling linguistic spatial terms. Languages (spoken or signed) have at their spatial core the ability to address not only the axiomatic building blocks *per se*, but to tie them together into larger groupings [45]. Spatial language can thus be explicit (as in the case

of a term like *disjoint*), or it can be vague (as in the case of *along*), leading to uncertainty [40]. Talmy [45] asserts that there seems to be a set of universal primitives that is approximately closed with which to construct concepts from; therefore, constructions like the 9-intersection and the Region Connection Calculus move semantic terms into rigorously defined mathematical terminology that come straight from topology. Natural-language spatial prepositions have been studied on a computational level [1], but not with the conceptual neighborhood graph backdrop, which would offer a rationale for relating most similar terms. A study of road-and-park relations showed that people link particular constructions to particular language terms, and these constructions—though topologically distinct from one another—are close to each other within the neighborhood graph [36,43]. When the constructions are separated within the graph, the separation is the by-product of prototypical relations, rather than a fundamental rift. A cognitively plausible conceptual neighborhood graph must keep the atomic relations that constitute a spatial disjunction connected to each other. For this purpose we examine the *convexity* of disjunctions of atomic relations.

Convexity has been applied in GIS for convex hulls [38], object decomposition and reconstruction [10], in the surveyor's formula [8], and in cell trees for geometric data storage [25]. It has made but one appearance in spatial domains pertinent to the conceptual neighborhood graph [4], which studied convex relations of a particular cardinality, based on pre-convex relations [34]. In the temporal domain, convexity has been a criterion to analyze a set of relations similar to the topological ones [2] and their conceptual neighborhood graphs [24] in order to assist in temporally interpreting prose and looking for descriptive consistency [42]. Each convex subgraph of the conceptual neighborhood has been placed into a subset/superset lattice to provide a temporal aggregate neighborhood. This paper investigates whether natural-language spatial disjunctions of the relations between two simple regions correspond to convex sets of the neighborhood graphs.

The side effects of convexity—shape and path redundancy—have important impacts on people's decision-making and certain spatial motor skills. Decision trees—a graphical structure where all connected subgraphs are convex—represent the optimal decision structure for experts [11]. Corroborative evidence (the mental manifestation of multi-path decisions) has been found to produce increases in learning [22], retention [5], argument construction [49], and memorization [50], and furthermore is an often-cited burden of proof in law [26,32]. On a motor skill level, as people learn to pick up objects at young ages, they acquire the skill of grasping the object as if it were convex, lending preference to that type of object [41]. Also, the judgment of position sees significant benefit from convexity [7]. While both spatial motor skills are dependent upon the convexity of a geometric solid or geometric scenario, the psychological impacts that would be mirrored in a graph structure lend favorably to considering its impact upon conceptual neighborhood graphs. Further evidence exists to support that reasoning itself is a spatial process [31], and convexity at its core is a spatial property (contrived through distance structure). This paper offers a platform for analyzing the conceptual similarities of spatial terms in any natural language based on the atomic region-region relations found at their hearts: an ordering of convex relations based on the conceptual neighborhood graph.

The remainder of this paper is structured as follows. Section 2 summarizes the underlying model for topological relations and their conceptual neighborhood graph.

Based on the formal definition of convexity in graphs (Section 3), we derive in Section 4 the complete set of convex subgraphs of the region-region relations on the surface of the sphere in their A-neighborhood graph [17]. Commonplace English spatial prepositions [33] are then mapped onto disjunctions of the region-region relations and are shown to be members of the set of convex subgraphs (Section 5). A lattice of the convex subgraphs is constructed (Section 6), which is used for translating terms between languages (Section 7). Section 8 draws conclusions and discusses future work.

2 Topological Relations and Conceptual Neighborhood Graphs

The binary topological relations between two simple regions—that is, regions that are homeomorphic to 2-discs—are the focus of this study. The 9-intersection [20] captures these relations through the pairwise intersections of two regions interiors, boundaries, and exteriors. Topological invariants of these nine intersections (i.e., properties that are preserved under topological transformations) categorize topological relations. The content invariant—distinguishing empty (\emptyset) and non-empty ($\neg\emptyset$) intersections—is the most generic criterion, as other invariants can be considered refinements of non-empty intersections. A 3x3 matrix captures these specifications concisely. Pairs of regions with different 9-intersection matrices have different topological relations. For regions embedded in \mathbb{R}^2 , eight different relations—called *disjoint*, *meet*, *overlap*, *equal*, *coveredBy*, *inside*, *covers*, and *contains*—can be distinguished with empty and non-empty intersections (Fig. 1). Another three 9-intersection matrices—called *attach*, *entwined*, and *embrace*—are found when the regions are embedded in \mathbb{S}^2 , the surface of the sphere [16].

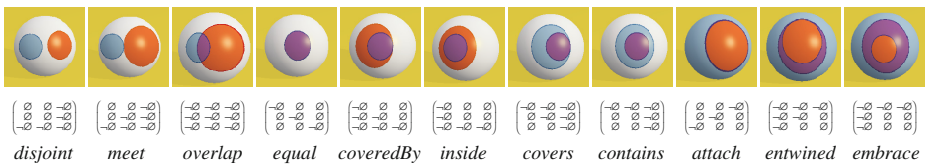


Fig. 1. The eleven topological relations between two regions embedded in \mathbb{S}^2 with the relations' 9-intersection matrices and their labels [16]

Since this set of relations is jointly exhaustive and mutually exclusive, exactly one of these eleven relations applies to any pair of simple regions. These base relations can be combined with an *exclusive disjunction* (XOR) to capture scenarios when a specific relation cannot be described. A total of 2^{11} different combinations are possible. The least specific case is called the *universal relation*, which is the exclusive disjunction of all eleven relations. The eleven cases with exactly one relation are the *atomic relations*. On the disjunctions and atomic relations, the *intersection* of relations applies, yielding in the case of no common relation the *empty relation* (i.e., a scenario that cannot be realized).

These topological relations *per se* are on a nominal scale. Although there is no strict order that applies to these relations, the eleven topological relations can be arranged such that pairs of most similar relations are grouped together [18], much like the arrangement of interval relations [24]. Different rationales for such similarity grouping lead to different arrangements of the relations, called *conceptual neighborhood graphs* [24]. Three basic types of neighborhood graphs are common—the A-neighborhood, which derives the similar relations from anisotropic scaling; the B-neighborhood (similarity from rotation or translation while preserving size and shape); and the C-neighborhood (similarity from isotropic scaling). Another five neighborhoods (Fig. 2) provide closure under union and intersection [17].

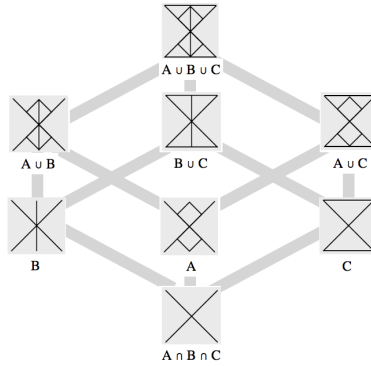


Fig. 2. The family of conceptual neighborhood graphs of the eleven topological relations between two regions embedded in \mathbb{S}^2 [17]

3 Convexity and Subgraphs

In this section, the mathematical underpinnings of the paper are provided to give rise to common terminological ground.

Definition 1. Let V be a set of vertices and E be a set of edges connecting $v_i, v_j \in V$. The construction $V \cup E$ is called a *graph* and is denoted $G_{V,E}$. A subset of a graph is referred to as a *subgraph* [3].

Definition 2. If a subgraph contains all edges that connect members of its vertex set, then the subgraph is called an *induced subgraph*.

Definition 3. Let $H_{A,B}$ be an induced subgraph from $G_{V,E}$. If for every $a_i, a_j \in A$, every shortest path connecting a_i to a_j within $G_{V,E}$ is found within $H_{A,B}$, $H_{A,B}$ is *convex* [3].

Theorem 1. Let t, u, v be vertices in a graph G such that a u - v path exists between each pair. Further let $d(r,s)$ represent the distance between a specified pair of vertices r and s . Vertex t can be found on a shortest path between u and v if and only if:

$$d(u,t) + d(t,v) = d(u,v). \tag{1}$$

Proof: Assume that t is on a shortest path between u and v . We must show that $d(u,t) + d(t,v) = d(u,v)$. Since t is on a shortest path, there can be no shorter path between u and t than the one of length $d(u,t)$ by construction. Similarly for $d(t,v)$. Since shortest paths cannot have loops, both $d(u,t)$ and $d(t,v)$ do not share a common vertex other than t . The distance $d(u,v)$ is minimized under this construction and is thus a shortest path. Any other vertex can only produce a value greater than or equal to this one. Now assume that $d(u,t) + d(t,v) = d(u,v)$. We must now show that t sits on a shortest path. Assume not. This implies that $d(u,t) + d(t,v) > d(u,v)$, which contradicts the initial assumption. Therefore t must be on a shortest path. ■

4 Application to Simple Region-Region Relations on the Sphere

Theorem 1 allows for reducing an algorithm for determining convex subgraphs to a shortest path calculation and a sequence of tests of additive distance. If the distance sum is found to fit the condition of Eqn. 1, a vertex t must be contained within a subgraph that contains vertices u and v for all possible t and all possible pairs u,v . If not, the subgraph cannot possibly be convex.

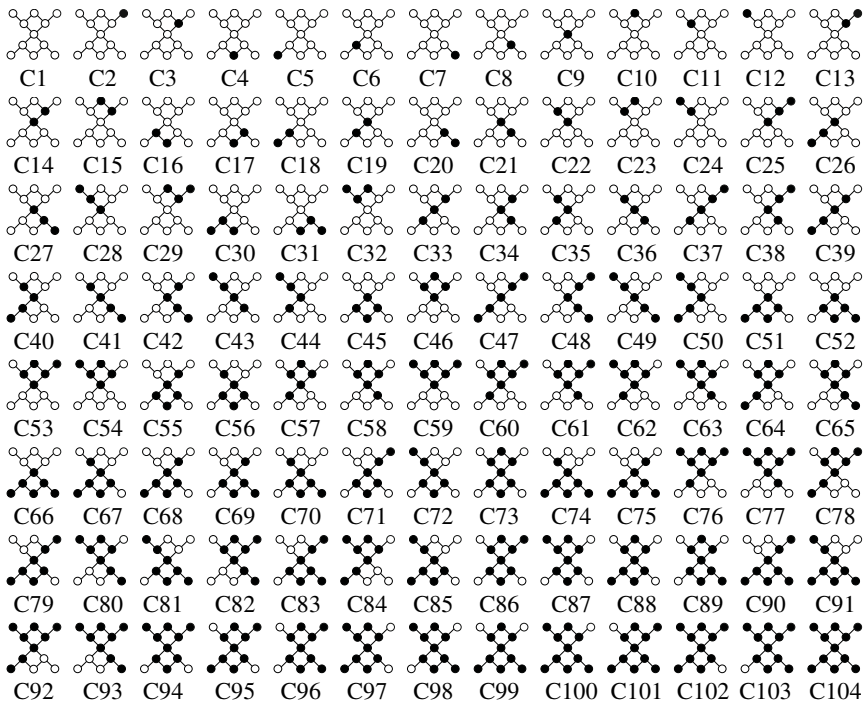


Fig. 3. The complete set of convex subgraphs of the A-neighborhood graph, ranging from C1 (empty disjunction) to C104 (universal relation) [35]

We implemented this algorithm and derived the complete set of convex subgraphs for the eleven region-region relations on the sphere [16] over their A-neighborhood graph. Among the 2^{11} subgraphs of the A-neighborhood, 104 (5%) are identified as convex (Fig. 3). The set of convex relations includes the empty relation (C1), the eleven atomic relations (C2–12), and the universal relation (C104).

5 Common Spatial Prepositions as Disjunctions of Topological Relations

From among three different perspectives of *cognitive plausibility*—(1) maintenance of human success and errors (typical of cognitive modeling), (2) inputs and outputs match human thought (typical of artificial intelligence), and (3) rational behavior (typical of social simulations) [27]—we are concerned particularly with the inputs and outputs of a machine system for inference. To get an adequate understanding of plausibility, we examine English-language terms that people typically use to describe region-region relations.

Landau and Jackendoff provided a “fairly complete list of the prepositional repertoire of English” [33] that applies to region objects. Among their 78 terms are two groups of terms that are outside the scope of this investigation. First, a large contingency of terms refers primarily to direction or orientation (e.g., above/below, beneath/on top of, in front/behind, up/down, left/right, north/south/east/west). Since the present focus is on topological relations, only terms without an explicit reference to direction are considered here. Second, a fair number of terms are intransitive or map onto relations that are not binary (e.g., here/there, between, among). Since topological relation are binary in nature, only those terms that apply to exactly two regions are considered. Finally, since some of the 78 terms are considered synonyms (e.g., along and alongside, inside and in, outside and out) only non-redundant terms are considered. After consolidation, a total of sixteen spatial prepositions remain for this investigation of convex topological relations (Table 1).

Table 1. Sixteen of the 78 spatial prepositions that describe region-region configurations without an explicit reference to direction or orientation

about	across	along	around	at	beside	by	far from
in	near	out	to the side of	together	through	throughout	with

The goal is to express these terms as disjunctions of the eleven 9-intersection relations, which then map onto subgraphs of the A-neighborhood graph. Given natural preferences for convex objects, we expect each of the sixteen spatial prepositions to have a construction from the set of 104 convex relations. Details of these mappings are given for the four predicates *outside*, *inside*, *crosses*, and *along* (Fig. 4). These terms in particular are used to describe many specific configurations of objects, and all of them abstract away a level of detail that is determined to be unimportant given the scenario. Given the nature of language, information systems have to be able to account for such abstraction mechanisms to accommodate spatial searching features.

While context can tell the human user a lot about which meaning is implied, the system itself has no such knowledge of the contextual basis for the term, leaving it in a position to conduct inference on an uncertain set of terms that are often found to be quite precise for the person accessing the information that system can provide.

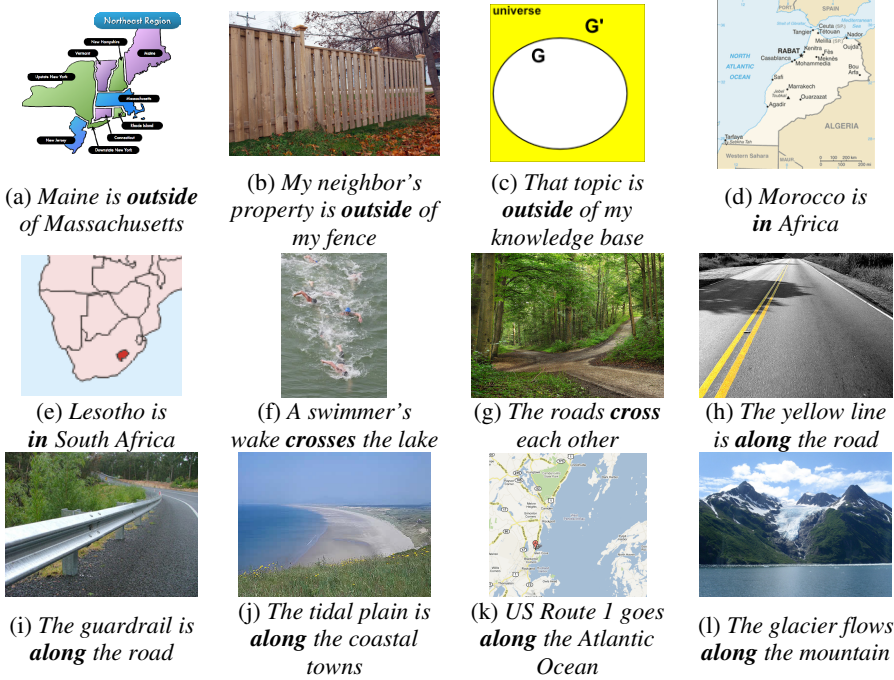


Fig. 4. Uses of spatial terms: (a-c) *outside* mapping respectively onto *disjoint*, *meet*, or *attach*; (d-e) *in* mapping respectively onto *covered by* and *inside*; (f-g): *crosses* mapping respectively onto *coveredBy* and *overlap*; and (h-l) *along* mapping respectively onto *inside*, *overlap*, *coveredBy*, *disjoint*, and *meet*.

Table 2 shows that each of the sixteen spatial prepositions indeed takes on the preferential form of convexity. While this pattern may be considered as proof enough, each of these subgraphs are of course *connected*, of which convexity is a special case. Many of the problems that people arrive at spatially when translating languages have to do with the presence of the explicit terms. Only four spatial words for planar regions relations—*in*, *out*, *far*, and *near*—can be found in the Natural Semantic Metalanguage [48], which indicates that few explicit terms are found across families of languages.

Topologically explicit terms map onto a single atomic relation (C2–C12). For the sixteen spatial prepositions, this condition applies only to *through* (C9) and to *far from* (C2). The remaining terms are *topologically ambiguous*. While explicit terms are preferable for communication, ambiguous terminology serves a fundamental purpose as well, as they are abstractions of things that do not matter in the grand scheme of conversation or context. The image created by ambiguous expressions is not impacted substantially, even if the explicit relation changes. For example,

someone asking whether two things are connected does not care how, but cares that one can get from one place to the other without leaving either territory. Whether or not the items are connected only at one point or one is completely inside the other makes no difference from that perspective.

Table 2. The sixteen spatial prepositions (Table 1) equated to explicit atomic constructors

Spatial Preposition	Union of Atomic Relations	Convex Relation
about	<i>equal, coveredBy, inside</i>	C30
across/crosses	<i>overlap, coveredBy</i>	C19
along	<i>disjoint, meet, overlap, coveredBy, inside</i>	C47
around	<i>disjoint, meet</i>	C13
at	<i>equal, coveredBy, inside</i>	C30
beside	<i>disjoint, meet, attach</i>	C29
by	<i>disjoint, meet, attach</i>	C29
far from	<i>disjoint</i>	C2
in/inside	<i>coveredBy, inside</i>	C18
near	<i>disjoint, meet, attach</i>	C29
out/outside	<i>disjoint, meet, attach</i>	C29
through	<i>overlap</i>	C9
throughout	<i>equal, covers, contains</i>	C31
to the side of	<i>disjoint, meet, attach</i>	C29
together	<i>overlap, equal, coveredBy, inside, covers, contains, entwined, embrace</i>	C91
with	<i>overlap, equal, coveredBy, inside, covers, contains, entwined, embrace</i>	C91

While the sixteen spatial prepositions prevail in natural English language, the domain of mathematics offers more spatial terms for technical usage. To bring them into the same framework, the mathematical terms *connected, equal, nothing, subset, superset, touching, unequal, and universal* are mapped onto corresponding atomic 9-intersection relations (Table 3).

Table 3. The seven mathematical spatial terms equated to explicit atomic constructors

Spatial Preposition	Union of Atomic Relations	Convex Relation
equal	<i>equal</i>	C4
nothing	\emptyset	C1
subset	<i>coveredBy, inside</i>	C18
superset	<i>covers, contains</i>	C20
touching	<i>meet, attach</i>	C15
unequal	<i>disjoint, meet, overlap, coveredBy, inside, covers, contains, attach, entwined, embrace</i>	–
universal	<i>disjoint, meet, overlap, equal, coveredBy, inside, covers, contains, attach, entwined, embrace</i>	C104

Two of these seven mathematical terms are explicit (*equal* and *nothings* map onto C4 and C1, respectively). Another four terms are convex; however, *unequal* is not a convex relation.

6 A Convex Ordering

Given that convex relations may in fact serve as links in many domains, one needs an understanding of the structure of the convex relations to one another, allowing for autonomous neighborhood graph generation by users in isolated fields. A prime example of this is the comparison of linguistically based neighborhoods that only reflect terminology that exists within a particular language. While spoken language may be descriptive enough for some, written, signed, and drawn languages may be able to convey more concepts [45].

The construction of an ordering needs a mechanism to sort the subgraphs. For neighborhood graphs in general, there are two typical approaches: physical deformations [16-18,21,24] and representational lattices [4,12,42], banking on the condition that topological deformations occur smoothly so that homeomorphic members will pass to a relation that is either a subset or superset under the 9-intersection matrix. In this disjunctive case, the representational lattice approach is taken, indicative of a “power set” construction (Fig. 5).

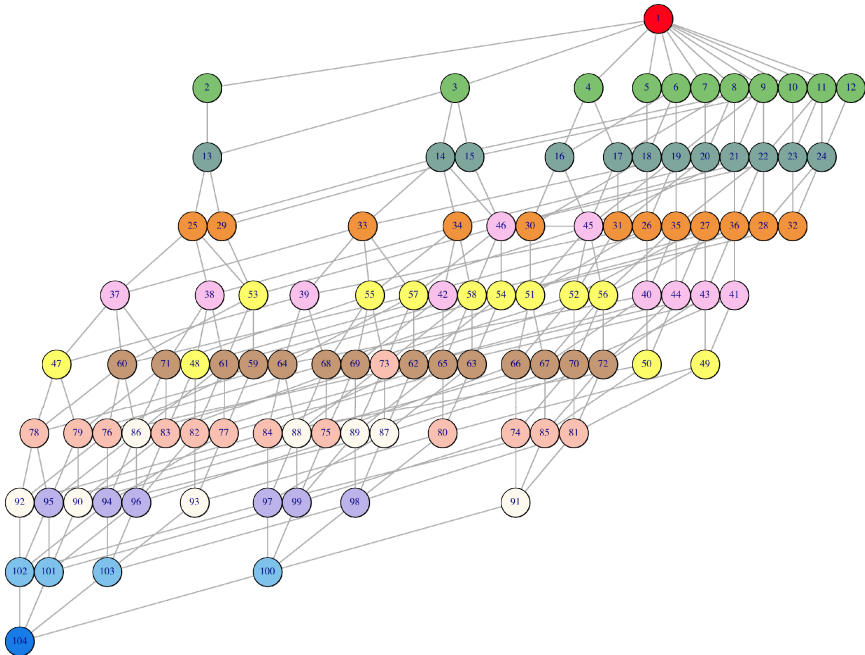


Fig. 5. Lattice of the 104 convex relations with different colors representing disjunctive sets of differing cardinalities. The graph is aligned in the optimal Reingold-Tilford construction.

As an example, *touching* (C15) is the union of the convex relations *meet* (C3) and *attach* (C10). It is also a subset of the convex relation *outside* (C29). These three terms mathematically are the most similar to the term *touching*. Any of the spatial prepositions presented here can be bounded by other linguistic terms in a similar manner.

Using the terms explored in Section 4, the lattice of convex relations can be exploited to provide a generalization/specification structure based on the terms themselves. Fig. 6 shows the ordering of the spatial prepositions from Table 2. While the example given here is for the English language, such a procedure can be employed on any language using the same principle. Comparing such word networks can point out significant differences in languages, either within the same language family or from differing families.

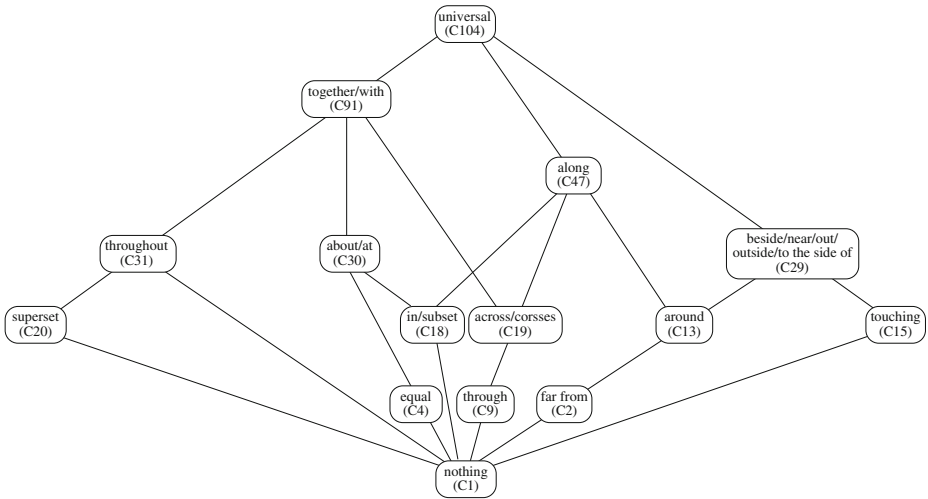


Fig. 6. The lattice of convex relations for the sixteen spatial prepositions and the six mathematical spatial terms

7 Translating an Unfamiliar Term

Given the implication from the Natural Semantic Metalanguage that the vast majority of particular spatial concepts do not exist in all languages, some languages contain terms that are unknown to other languages. In creating information systems, it is plausible to assume that entries will be made in varying languages or in translations from one language to another, creating the necessity for addressing terms without a direct language parallel within two or more contributing languages. The lattice of convex relations (Fig. 6) enables a straightforward mathematical translation of one term to another term. More involved, however, is an automated language translation to invoke a suitable understanding of the term, using only a user's familiar lexicon. This translation is developed subsequently.

Borrowing from the idea that any object can be constructed as a union of convex objects [10], an algorithm is designed to provide a minimal precise cover of the graph representing the foreign term, and use the minimal set as a disjunctive description. To illustrate this concept, consider the statement: *sensors detected rain north of Zanzibar*.

While the concept of *north* is foreign to the current lexicon (Fig. 6), because it is directionally charged, itself not a topological property. The term does, however, bring with it some implied topological information that can be used. The obvious example of *north* is the way that Kenya is related to Zanzibar (i.e., separated from one another and directly “above”), referring to *disjoint*. On the other hand, Kenya relates to Tanzania (of which Zanzibar is a part) as Kenya and Tanzania *meet*. One can also think of it by a latitudinal perspective: it is raining at all points with greater latitude than Zanzibar, precisely the scenario of *attach*. Those three possibilities represent the term *outside*. If it is raining on the island of Zanzibar and it is raining in Kenya and over the open water between them, this scenario represents *overlap*. These four atomic relations—*disjoint*, *meet*, *attach*, and *overlap*—together all contribute to our understanding of *north of* (some more than others), but all have instances that would be expressed under that paradigm. The set—though a union of convex sets—is not a convex set itself (Fig. 7a), as it is missing at least one other possibility, *entwined*. If in the *attach* scenario precipitation slightly penetrates the island of Zanzibar, one crosses over the line from *attach* to *entwined*. Similarly, *embrace* could be added on to the set. The additions of *entwined* (Fig. 7b) or *entwined* and *embrace* (Fig. 7c) yield convex relations.

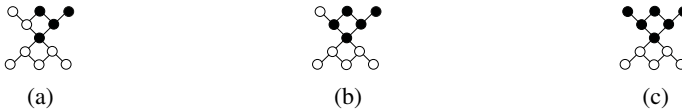


Fig. 7. The set of base relations *disjoint*, *meet*, *overlap*, and *attach* (a) is not convex, (b) becomes convex with the addition of *entwined* (C53), and (c) becomes convex with the addition of *entwined* and *embrace* (C59)

The algorithm `Translate` is developed to create a minimal disjunctive phrase (`knownTerms`) for an unfamiliar term (`foreignTerm`), which has been seed with its atomic relations (`foreignAtoms`). It traverses recursively the lattice of known terms (e.g., Fig. 6) starting at its root `C104` and tests at each node whether its atomic relations (`allAtomicRels`) are fully included in the seeded atomic relations of the foreign term. If so, that node’s terms are added to the disjunctive phrase and no more detailed terms are needed; otherwise, all descendants of that node are examined recursively with the same procedure. To eliminate possible false hits for nodes with multiple parents, a final test is needed to clean up any terms for which a more generic term (i.e., in the lattice $\text{convexRel}(t1) < \text{convexRel}(t2)$) is included as well. While not a direct translation, the algorithm creates an understandable image of possibilities in the smallest construction possible.

```

Algorithm Translate (foreignTerm, foreignAtoms, knownTerms)
var n: node, t1, t2: term
begin
knownTerms ← ∅
n ← C104 %root of lattice
translate1 (foreignTerm, foreignAtoms, knownTerms, n)
for each term t1 in knownTerms do
  for each term t2•t1 in knownTerms do
    if convexRel(t1)<convexRel(t2) then knownTerm ← knownTerm \ t1
end.

```

```

Translate1 (foreignTerm, foreignAtoms, knownTerms, n)
var d: node
begin
for all descendents d of n do %traverse lattice
  if d<>C1 then %excludes the empty convex relation
    if allAtomicRels(d)  $\subseteq$  foreignAtoms then
      setOfKnownTerms  $\leftarrow$  setOfKnownTerms  $\cup$  terms(d)
    else translate1(foreignTerm, foreignAtoms, knownTerms, d)
end.

```

For instance, the term *unequal* (which resulted in a non-convex relation) can now be translated into a minimal disjunction of known terms. Seeding *unequal* with the disjunction of ten terms (Table 2), `translate` generates the description “superset or in/subset or along, or beside/near/out/outside/to the side of.”

8 Conclusions and Future Work

In this paper, we have argued for convexity as a relevant property to study within the spatial domain of conceptual neighborhood graphs, providing a potential glimpse into cognitive plausibility based on an artificial intelligence perspective. We introduced a formal definition of convexity in graphs, which allowed us to identify all 104 of the 2,048 possible induced subgraphs of the A-neighborhood of simple region-region relations on the sphere as convex subgraphs, representing 5% of the possibilities.

Though 5% is relatively rare, we analyzed sixteen English spatial prepositions (the subset of the 78 terms [33] that does not have a primary direction or orientation dependency) and seven mathematical spatial terms that describe the relations between two planar regions. We found that all sixteen natural-language prepositions were members of the convex relation set. Only one mathematical term—*unequal*—was not convex. Since *unequal* is essentially the negation of *equal*, it is the only term that describes a configuration in negative terms. This finding is in line with the insight that people primarily encode information based on positive and present information [23].

The result about the convexity of all positive terms served as motivation for the construction of a lattice of convex relations (similar to the temporal convex relations [42]) that allows for differing languages to construct semantic neighborhoods based on vocabulary common to their speakers and to integrate the process of translating prose data into a unified spatial system language. Based on the lattice of spatial terms we developed an algorithm that generates for a spatial term that is not part of a lexicon a minimal disjunction of known terms. This can pave the way for context-aware systems [17], understandings [44], and neighborhood graphs [13].

There are many directions to go with convex analysis within the settings of the closed spatial algebras that we understand such as the Region Connection Calculus, the 9-intersection, and the Qualitative Trajectory Calculus [47]. While this study focused only on the A-neighborhood of the eleven region-region relations, the two other basic conceptual neighborhoods, as well as their unions, should be assessed in a similar way in order to identify their merit. The rich corpus of spatial predicates for line-region relations from the road-and-park study [43] offers another path to assess the importance of convex relations. While composition of topological relations has

been studied extensively in the past, the composition of disjunctions of relations [6,46] is uncharted territory. Analyzing the results of these compositions can provide valuable insight into the driving forces of connectivity and convexity in the quest of humans to deal with uncertainty on a spatial and temporal level. Early results indicate that the overwhelming majority of compositions of disjunctions in the 9-intersection model result in convex relations, but that figure is sufficiently biased by compositions that net no further information than universal, itself a convex set.

Acknowledgments. This work was partially supported by NSF Grants IIS-1016740 (PI: Max Egenhofer) and DGE-0504494 (PI: Kate Beard). Special thanks to Brian Lopez-Cornier, an Upward Bound Math and Science student from Lawrence High School in Lawrence, MA, for his work on the preliminary construction of the convexity algorithm.

References

1. Abella, A., Kender, J.: Qualitatively Describing Objects Using Spatial Prepositions. In: IEEE Workshop on Qualitative Vision, pp. 33–38. IEEE Computer Society, Washington (1993)
2. Allen, J.: Maintaining Knowledge about Temporal Intervals. *Communications of the ACM* 26(11), 832–843 (1983)
3. Artigas, D., Dourado, M., Szwarcfiter, J.: Convex Partitions of Graphs. *Electronic Notes in Discrete Mathematics* 29, 147–151 (2007)
4. Balbiani, P., Condotta, J.-F., Fariñas del Cerro, L.: A Tractable Subclass of the Block Algebra: Constraint Propagation and Preconvex Relations. In: Barahona, P., Alferes, J.J. (eds.) EPIA 1999. LNCS (LNAI), vol. 1695, pp. 75–89. Springer, Heidelberg (1999)
5. Beers, G., Bowden, S.: The Effect of Teaching Method on Long-term Knowledge Retention. *Journal of Nursing Education* 44(11), 511–514 (2005)
6. Bennett, B.: Some Observations and Puzzles about Composing Spatial and Temporal Relations. In: Rodríguez, R. (ed.) ECAI 1994 Workshop on Spatial and Temporal Reasoning (1994)
7. Bertamini, M.: The Importance of Being Convex: an Advantage for Convexity when Judging Position. *Perception* 30, 1295–1310 (2001)
8. Braden, B.: The Surveyor’s Area Formula. *The College Mathematics Journal* 17(4), 326–337 (1986)
9. Bruns, T., Egenhofer, M.: Similarity of Spatial Scenes. In: Kraak, M., Molenaar, M. (eds.) Seventh International Symposium on Spatial Data Handling, pp. 31–42 (1996)
10. Bulbul, R., Frank, A.: Intersection of Nonconvex Polygons Using the Alternate Hierarchical Decomposition. In: Painho, M., Santos, M., Pundt, H. (eds.) Geospatial Thinking, pp. 1–23. Springer, Berlin (2010)
11. Busemeyer, J., Weg, E., Barkan, R., Li, X., Ma, Z.: Dynamic and Consequential Consistency of Choices Between Paths of Decision Trees. *Journal of Experimental Psychology: General* 129(4), 530–545 (2000)
12. Dube, M.: An Embedding Graph for Topological Spatial Relations, Master’s Thesis. University of Maine (2009)
13. Dube, M.P., Egenhofer, M.J.: Establishing Similarity across Multi-granular Topological-Relation Ontologies. In: Rothermel, K., Fritsch, D., Blochinger, W., Dürr, F. (eds.) QuaCon 2009. LNCS, vol. 5786, pp. 98–108. Springer, Heidelberg (2009)

14. Egenhofer, M.J.: A Model for Detailed Binary Topological Relationships. *Geomatica* 47(3), 261–273 (1993)
15. Egenhofer, M.J.: Deriving the Composition of Binary Topological Relations. *Journal of Visual Languages and Computing* 5(2), 133–149 (1994)
16. Egenhofer, M.J.: Spherical Topological Relations. *Journal on Data Semantics III*, 25–49 (2005)
17. Egenhofer, M.J.: The Family of Conceptual Neighborhood Graphs for Region-Region Relations. In: Fabrikant, S.I., Reichenbacher, T., van Kreveld, M., Schlieder, C. (eds.) *GIScience 2010. LNCS*, vol. 6292, pp. 42–55. Springer, Heidelberg (2010)
18. Egenhofer, M.J., Al-Taha, K.: Reasoning about Gradual Changes of Topological Relationships. In: Frank, A.U., Formentini, U., Campari, I. (eds.) *GIS 1992. LNCS*, vol. 639, pp. 196–219. Springer, Heidelberg (1992)
19. Egenhofer, M.J., Franzosa, R.: Point-set Topological Spatial Relations. *International Journal of Geographical Information Systems* 5(2), 161–174 (1991)
20. Egenhofer, M.J., Herring, J.: Categorizing Binary Topological Relationships Between Regions, Lines, and Points in Geographic Databases, Department of Surveying Engineering, University of Maine, Orono, ME (1991)
21. Egenhofer, M.J., Mark, D.: Modeling Conceptual Neighborhoods of Topological Line-Region Relations. *International Journal of Geographical Information Science* 9(5), 555–565 (1995)
22. Emig, J.: Writing as a Mode of Learning. *College Composition & Communication* 28, 122–127 (1977)
23. Freeman, J.: The Modeling of Spatial Relations. *Computer Graphics and Image Processing* 4, 156–171 (1975)
24. Freksa, C.: Temporal Reasoning based on Semi-Intervals. *Artificial Intelligence* 54(1), 199–227 (1991)
25. Gunther, O.: The Design of the Cell Tree: an Object-Oriented Index Structure of Geometric Databases. In: *Fifth International Conference on Data Engineering*, pp. 598–605. IEEE Computer Society, Washington, DC (1989)
26. Jones, L.: Corroborating Evidence as a Substitute for Delivery in Gifts or Chattels. *12 Suffolk University Law Review* 16 (1978)
27. Kennedy, W.: Cognitive Plausibility in Cognitive Modeling, Artificial Intelligence, and Social Simulation. In: Howes, A., Peebles, D., Cooper, R. (eds.) *9th International Conference on Cognitive Modeling*, pp. 454–455 (2009)
28. Klippel, A., Li, R., Yang, J., Hardisty, F., Xu, S.: The Egenhofer-Cohn Hypothesis: Or, Topological Relativity? In: Raubal, M., Frank, A., Mark, D. (eds.) *Cognitive and Linguistic Aspects of Geographic Space—New Perspectives on Geographic Information Research* (in press)
29. Klippel, A., Li, R., Hardisty, F., Weaver, C.: Cognitive Invariants of Geographic Event Conceptualization: What Matters and What Refines? In: Fabrikant, S.I., Reichenbacher, T., van Kreveld, M., Schlieder, C. (eds.) *GIScience 2010. LNCS*, vol. 6292, pp. 130–144. Springer, Heidelberg (2010)
30. Klippel, A., Li, R.: The Endpoint Hypothesis: A Topological-Cognitive Assessment of Geographic Scale Movement Patterns. In: Hornsby, K.S., Claramunt, C., Denis, M., Ligozat, G. (eds.) *COSIT 2009. LNCS*, vol. 5756, pp. 177–194. Springer, Heidelberg (2009)
31. Knauff, M., Strube, G., Jola, C., Rauh, R., Schlieder, C.: The Psychological Validity of Qualitative Spatial Reasoning in One Dimension. *Spatial Cognition and Computation* 4(2), 167–188 (2004)

32. Koch, R.: Process v. Outcome: The Proper Role of Corroborative Evidence in Due Process Analysis of Eyewitness Identification Testimony. 88 *Cornell Law Review* 1097 (2003)
33. Landau, B., Jackendoff, R.: What and Where in Spatial Language and Spatial Cognition. *Behavioral and Brain Sciences* 16, 217–265 (1993)
34. Ligozat, G.: Tractable Relations in Temporal Reasoning: Pre-convex Relations. In: *European Conference on Artificial Intelligence: Workshop on Spatial and Temporal Reasoning*, pp. 99–108 (1994)
35. Lopez-Cornier, B., Dube, M.: An Algorithm for Determining Convexity within an Arbitrary Network. *Upward Bound Math and Science Journal of Explorations* (2011)
36. Mark, D., Egenhofer, M.J.: Modeling Spatial Relations Between Lines and Regions: Combining Formal Mathematical Models and Human Subjects Testing. *Cartography and Geographical Information Systems* 21(3), 195–212 (1994)
37. Nedas, K., Egenhofer, M.J.: Spatial-Scene Similarity Queries. *Transactions in GIS* 12(6), 661–681 (2008)
38. Preparata, F., Hong, S.: Convex Hulls of Finite Sets of Points in Two and Three Dimensions. *Communications of the ACM* 20(2), 87–93 (1977)
39. Randell, D., Cui, Z., Cohn, A.: A Spatial Logic Based on Regions and Connection. In: *Third International Conference on Knowledge Representation and Reasoning*, pp. 165–176 (1992)
40. Regier, T.: *The Human Semantic Potential: Spatial Language and Constrained Connectionism*. MIT Press, Cambridge (1996)
41. Santello, M., Soechting, J.: Gradual Molding of the Hand to Object Contours. *Journal of Neurophysiology* 79, 1307–1320 (1998)
42. Schilder, F.: A Hierarchy for Convex Relations. In: *Fourth International Workshop on Temporal Representation and Reasoning*, pp. 86–93 (1997)
43. Shariff, A., Egenhofer, M.J., Mark, D.: Natural-Language Spatial Relations Between Linear and Areal Objects: The Topology and Metric of English-Language Terms. *International Journal of Geographical Information Science* 12(3), 215–246 (1998)
44. Sjøo, K.: *Functional Understanding of Space: Representing Spatial Knowledge Using Concepts Grounded in an Agent's Purpose* (Ph.D. Dissertation), KTH Computer Science and Communication (2011)
45. Talmy, L.: The Representation of Spatial Structure in Spoken and Signed Language. In: Emmorey, K. (ed.) *Perspectives on Classifier Constructions in Sign Language*, Mahwah, NJ, pp. 169–195 (2003)
46. Tarski, A.: On the Calculus of Relations. *Journal of Symbolic Logic* 6, 73–89 (1941)
47. Van de Weghe, N., Kuijpers, B., Bogaert, P., De Maeyer, P.: A Qualitative Trajectory Calculus and the Composition of Its Relations. In: Rodríguez, M.A., Cruz, I., Levashkin, S., Egenhofer, M. J. (eds.) *GeoS 2005. LNCS, vol. 3799*, pp. 60–76. Springer, Heidelberg (2005)
48. Wierzbicka, A.: *Semantic Primitives*. Frankfurt, Athenäum (1972)
49. Wiley, J., Voss, J.: Constructing Arguments from Multiple Sources: Tasks that Promote Understanding and not Just Memory for Text. *Journal of Educational Psychology* 91(2), 301–311 (1999)
50. Zechmeister, E., McKillip, J., Pasko, S., Bepalec, D.: Visual Memory for Place on the Page. *Journal of General Psychology* 92(1), 43–52 (1975)

Toward Web Mapping with Vector Data

Julien Gaffuri

European Commission - Joint Research Centre
Via Enrico Fermi 2749, 21027 Ispra, Italy
julien.gaffuri@gmail.com

Abstract. Improving the use of vector data in web mapping is often shown as an important challenge. Such shift from raster to vector web maps would open web mapping and GIS to new innovations and new practices. The main obstacle is a performance issue: Vector web maps in nowadays web mapping environments are usually too slow and not usable. Existing techniques for vector web mapping cannot solve alone the performance issue. This article describes a unified framework where some of these techniques are integrated in order to build efficient vector web mapping clients and servers. This framework is composed of the following elements: Specific formats for vector data and symbology, vector tiling, spatial index services, and generalization for multi-scale data. A prototype based on this framework has been implemented and has shown satisfying results. Some principles for future standards to support the development of vector web mapping are given.

Keywords: Web mapping, standard, spatial data infrastructure, geo-portal, vector data, vector tiling, generalization, spatial index.

1 Introduction

An always increasing part of the maps we use every day are digital maps published on the Internet. If the first web maps were simple static images, web maps have progressively been considered as special images displayed within specific viewers. In such viewers, specific cartographic tools are available to explore the geographical space by panning and zooming in and out. Data layers from different servers can also be selected to be overlaid. The Internet has deeply changed the way maps are nowadays designed and used [1].

However, it seems the limit of existing web mapping technologies has been reached. To open a next level of interactivity and improve the user experience of web maps, it may be necessary to change the approach web maps are made with. This next step could be to enable a direct interaction of the user with the map objects. This interaction is not possible nowadays because web maps are, for a huge majority of them, based on raster data. Like paper maps, these maps are just images of objects the user can only see and not touch and manipulate.

The solution to go further in web mapping interactivity is to fully open web mapping to vector data. Vector data are nowadays mainly used in web mapping to build static raster maps to be published on a server. Developing a new web

mapping architecture to enable the publication and on the fly display of vector data would be an important step toward a new generation of web mapping applications.

In this paper, some benefits and challenges of shifting from raster to vector web mapping are presented. A state of the art of existing techniques for vector web mapping is given. Then, a framework unifying some of these techniques to make vector web mapping feasible is proposed. This framework integrates the following elements: Vector tiling, spatial indexing, multi-scale data and generalization. Finally, requirements for future vector web mapping standards are given.

2 Benefits and Challenges of Vector Web-Mapping

Improving the use of vector data in web mapping is often shown as the next challenge of web mapping [2,3,4]. Such change would allow, for example, unlocking the development of the following applications:

- A user could easily retrieve thematic and semantic information for each map object, like in a traditional GIS software. This information could be displayed in a specific window or a tool-tip. The user may have access not only to the primary attributes of the object but also to a wider set of external data linked to this object. This feature is especially important for augmented reality applications [5].
- Using the object geometries, some simple geoprocesses could be performed on the client side, like for example, computing the length of a road or the area of a parcel. To go further, more complex geoprocesses may be run on more advanced clients, which may open the gate to the fusion of web mapping and web GIS.
- Because the objects are rendered on the fly on the client, vector web mapping would allow an improved map content personalization. This personalization could be done at the layer level (the user may define his own style for a full data layer) and also at the object level (the user could make an important object bigger and display it with a different style).
- Many advanced digital cartography methods such as graphic generalization [6], label placement [7], legend customization [8,9], etc. may be introduced in web mapping clients. These modules may ensure the data rendering follows some basic cartographic principles. They may be loaded dynamically depending on the user needs.
- A true integration of data coming from different servers would become possible. Instead of having “lazy mash-ups”, where data layers of different servers are simply overlaid on top of each other, “smart mash-ups” could be developed. In such mash-ups, explicit relations between the objects may be computed and used for specific purposes. For example, a map showing pizzerias close to metro stations could be built from the integration and analysis of restaurant and public transport data layers.
- The interaction between data users and data providers may be improved. The users’ feedback on the data could be more explicit: Instead of specifying

only the location of an error described in a free text field, the user may submit a full and more precise update of the data. He could easily modify, add and delete objects. He could also capture new object geometries and snap them on existing geometries. This feature could support the development of collaborative maps and VGI [10].

- By opening web mapping to vector data, it would not be necessary anymore to pre-process and cache raster tiles. This may shorten the publication of updates of existing data. This is especially important to encourage the mapping of “live” data, like sensor data or geoRSS data. Nowadays, spatial dynamics are usually shown on videos records - only rarely raw live data are accessible to be displayed.
- The on the fly rendering of vector data by the client makes possible the development of new innovative cartographic visualization techniques (see for example figure 1), especially dynamic visualizations with moving and changing objects. Depending on the specific context of the user, and the nature of the data he wants to display, suitable cartographic visualization techniques may be developed.
- and finally, vector web mapping may certainly open many other advanced and innovative applications we cannot imagine yet [11].

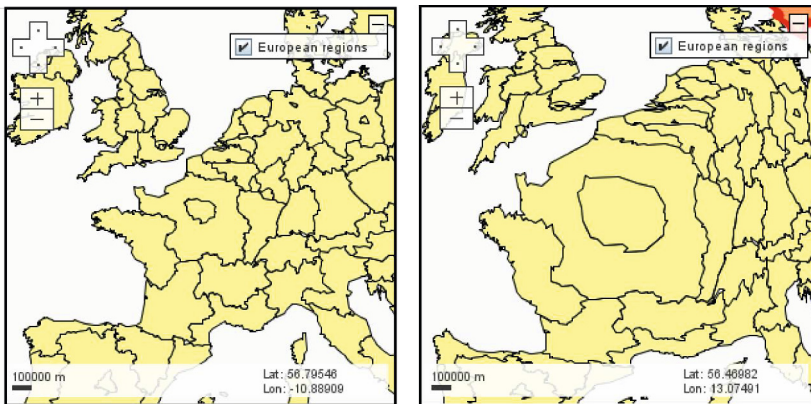


Fig. 1. Magnified view as described by [12]. On the right image, the map is magnified at the center of the view. A deformation of the vector data is computed on the fly when the user pans. See on-line demo: <http://www.opencarto.goldzoneweb.info/index.php?id=european-regions>.

The main obstacle to the development of vector web mapping is performance. Web maps must be fast maps, and existing web maps based on vector data usually do not meet the minimal requirements in terms of display speed. For this reason, the approach based on the publication of pre-computed raster maps has been preferred until now. However, taking into account that:

- client device memory, processing and connection capacities are always improving,
- and digital mapping methods of vector data, like generalization, are nowadays mature,

web mapping with vector data is becoming an acceptable approach. There are already emerging practices for web mapping systems based on vector data. If initiatives exist to make the shift to vector web mapping, it is not so common and nowadays, a huge majority of the map used on the Internet are raster maps. Rarely, vector data layers composed of usually few markers are displayed on top of raster maps.

One reason of this under-utilization of vector data is the lack of well-established, standardized and integrated approach to support efficient vector web mapping. Existing framework have been mainly developed for raster data and do not take into account the specific requirements of vector data. However, approaches exist to improve vector web mapping.

3 Approaches for Vector Web Mapping

The predominant approach to use vector data in web mapping is to extend existing raster clients to vector data. The client usually downloads vector data and displays it on top of raster images. The well-known limit of this approach is the long time usually necessary to transfer, decode and render the vector data. Furthermore, the final map is often not even legible because too dense for the map scale (see for example figure 2). As a result, the user waits a long time before an illegible map is displayed, and the application often becomes slow.

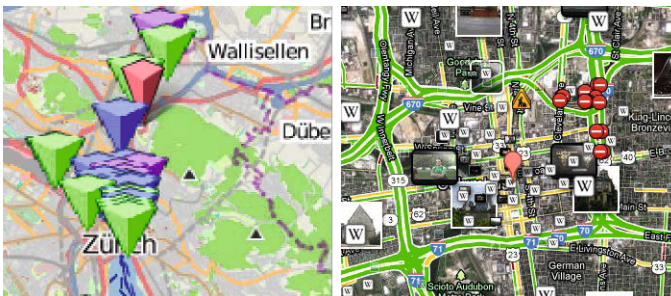


Fig. 2. Examples of existing web maps based on vector data

Approaches exist to improve the performance of web vector maps:

Use of specific data formats. The transfer duration is improved by the use of small and compressed formats for vector data and their symbology. There are many formats for vector data, most of them used in GIS softwares. A significant

part of them is based on XML [13,14] like KML, SVG, GML and SLD. XML based formats are efficient for spatial data exchange, but usually too verbose for a fast transmission, as required for vector web mapping. Some formats have been developed specifically for this purpose, like the GeoJSON format. Some vector formats allow to describe the object properties either as a list of (key,value), following the GIS practice, either embedded within HTML code, like in KML. File compression also helps making the files smaller (like zip compression for KMZ files, and several JSON compressions for GeoJSON). Beside vector data formats, style formats allow to describe how vector data are rendered. In some vector formats like SVG and KML, the styles are encoded within the data file. Some other formats like geoCSS, GSS and SLD allow an independent encoding of the data and their associated styles.

Vector tiling. Existing vector web mapping applications often load a full file containing vector data the user will never see, because outside of its current view. Vector tiling [15,16,17] allows to ensure only the data within the user's view are requested and loaded by the client. The principle is to decompose the vector dataset into different parts, each of them corresponding to vector data contained within a tile (see figure 3). In the case vector objects belong to several tiles, these objects are cut into pieces and each piece is assigned to the corresponding tile. Only the tiles are published on the server (usually one file per tile) and the client requests, caches and renders only the suitable tiles depending on its view and zoom level. Useless data outside of the view are not retrieved, which allows a performance improvement. A drawback of this method is the necessity to reassemble the objects on the client side. Compared to raster tiling, vector tiling is relatively new in web mapping and not well established yet.

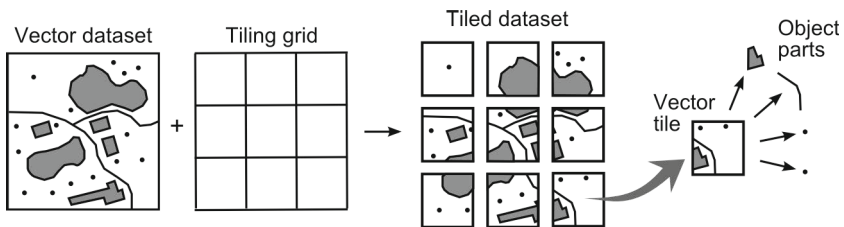


Fig. 3. Principle of vector tiling

Multi-scale data and generalization. The performance problem in vector web mapping is often due to the use of too detailed vector data. Indeed, such data are cumbersome to transfer, load and render, and may also not be legible as shown on figure 2. The solution is to provide to the client vector data with a level of detail suitable with the chosen zoom level. When the zoom level changes, new vector data with a suitable level of detail for this zoom level are requested, cached

and rendered. For this purpose, a multi-scale vector database is required on the server. A multi-scale database provides different representations of a region with different levels of details. Such multi-scale database can be produced automatically by generalization. Generalization has been identified as one of the key elements to make vector web mapping possible [18,19]. Its automation is known as a challenging issue, and has been the topic of many research for years [20,21]. Generalization is nowadays well formalized and operational techniques are used to automate many data and map production processes. In web-mapping, mainly the Ramer-Douglass-Peucker filtering algorithm and some clustering algorithms are used. Richer generalization methods exist [20,21] but have, surprisingly, not been adopted in web mapping.

Progressive transmission. Progressive transmission and streaming methods exist for many kind of data, like images [22]. Specific methods have been developed for vector data [23,24,25,26,27,28,29]. The principle of these methods is to load progressively the points composing the object geometries, and display the loaded data continuously, before the full transmission is complete. As a result, the data are displayed starting with a simplified view progressively enriched with additional details. Progressive transmission do not contribute to solve the performance problem. It improves the user experience but is not the prior aspect to focus on to unlock the use of vector data in web mapping. A progressive loading of the data may also be obtained using asynchronous queries to the server for each vector object.

None of the previous approaches allows to solve alone the performance problem – an efficient vector web mapping demands to use several together. In the next section we propose a framework that integrates some of these approaches and may help to progress toward vector web mapping.

4 An Integrated Framework for Vector Web Mapping

4.1 The Relevant “Data Slice”

The performance issue can be solved by serving **only** the relevant data to the user’s client. For this purpose, we propose to extract and serve the relevant “data slice” in the location-LoD space, as shown on figure 4. This relevant data slice depends on the selected position in the geographical space, and the selected zoom level [28]. Data outside of the view, and more detailed than what the zoom level requires are useless. Vector web mapping servers should send only these extracted data to the clients. This requirement illustrates that scale has to be considered as a full dimension of geographical information, like the three spatial dimensions and the temporal dimension [30]. For the selected zoom level, data with a relevant level of detail have to be provided. Furthermore, taking into account that:

- the viewer screen size (usually) do not change,
- and according to the equal information density law [31], the information density has to be constant whatever the zoom level,

all data slices should have comparable sizes, whatever the position and the zoom level. Consequently, depending on the client capacity (network bandwidth, memory, processing, screen size), a threshold data slice size should be defined, and the data slice provided by the server should not exceed this threshold size. The only way to ensure all data slices do not exceed this threshold size is to simplify the data by generalisation. The performance is controlled by the level of generalisation of the vector data: If the client faces performance issues, it means the vector data have not been simplified/generalised enough.

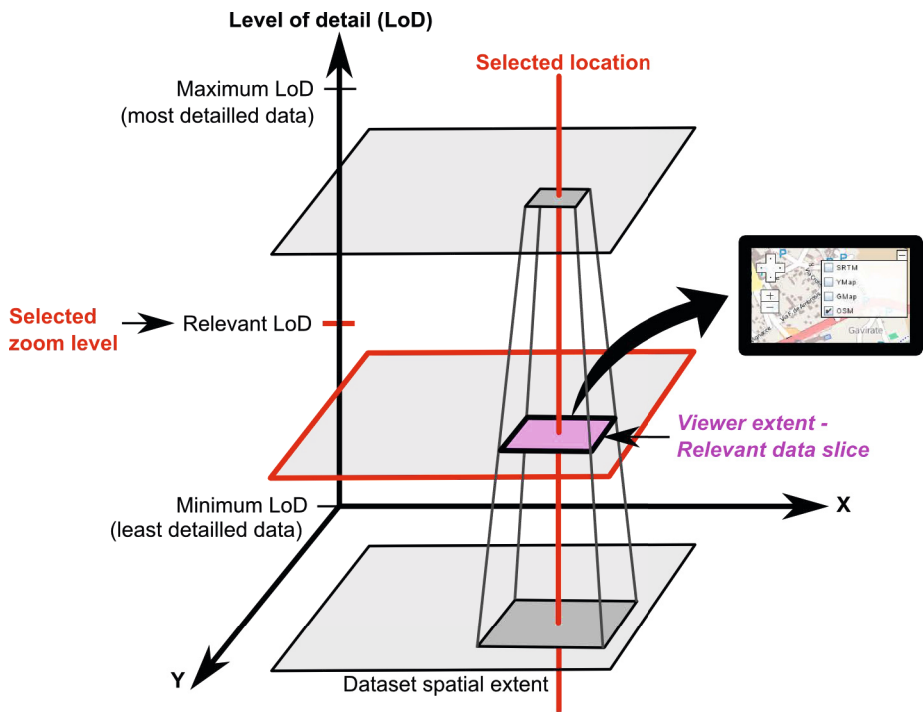


Fig. 4. The relevant information corresponds to the selected spatial extent, for the selected zoom level

In order to improve the performance of vector web mapping, it is necessary to extract the relevant data on both dimensions: Location and level of detail (LoD).

4.2 Location-Based Data Extraction

To extract the relevant data according to the view location, vector tiling and spatial indexing are used.

Vector Tiling. Vector tiling allows an efficient extraction of the relevant data according to the view location. Vector tiles are pre-computed on the server and identified according to their position in the tiling grid. Traditional tiling grids used for raster tiles may be applied also for vector tiles. The client needs the capability to retrieve vector tiles according to the view location, like in raster tiling. The following elements are also necessary:

- Vector objects are identified. The objects are built by merging their pieces belonging to different tiles and having a same identifier. The vector format used should include the possibility to identify the objects.
- The client is able to compute a union algorithm to build the object geometries from the union of their pieces. This union algorithm is however more simple than a generic union algorithm, taking into account that the geometries to union do not overlap and only touch each other along the grid lines. For linear geometries, this union is a simple concatenation of vertice lists. This union algorithm may be improved by including a code to each piece, that show from which side of the tile the original geometry has been cut. Further work may be undertaken to design such specific union algorithm for vector tiling.
- The client is able to cache vector data. Like for raster data, this caching improves the efficiency, even if it requires some memory capacities. For vector caching, two caches are required: For the tiles and for the vector objects.
- Object geometries and attributes are retrieved separately. Indeed, it is not necessary to retrieve the object attribute values for each object piece. A separation of both geometrical and semantic data enables to retrieve the object attribute values only once and improve the performance.

Spatial Indexing. Spatial indexing is a well-known technique in GIS to improve the location-based retrieval of vector objects. We propose to introduce spatial index services to improve the vector web mapping performance. Such service has the following characteristics:

- The spatial index structure is known by the client. We propose to use a quad-tree spatial index build on the same structure as the vector tiling grid.
- The spatial index service has the capability to provide:
 - References to the objects contained within a specified index cell,
 - an individual object from its reference.

The vector data retrieval is performed in two steps: First the client computes the relevant index cells depending on the view. If some of these cells have not been cached yet, the client sends a query to the spatial index service to retrieve the

references to the objects the cells intersect. Then, the client retrieves the object it has not cached yet - because an object may be referenced in several index cells, it may be already have been retrieved. As for vector tiling, two caches are needed: For the index cells and for the vector objects.

Vector Tiling or Spatial Indexing? Vector tiling and spatial indexing are two different strategies to do the same thing: Retrieve vector objects efficiently according to their location. None of these strategies is better. In the first one, the objects have to be reassembled on the client side. In the second one, two steps are required, and the whole object geometry is retrieved even if only one small part is within the view. The most suitable strategy depends on the kind of vector data: **Vector tiling is suitable for large and non compact object layers** (like for example contour lines, routes, GPS traces, etc.), while **spatial indexing for layers composed of small and compact objects** (like point objects, small areas, etc.). In case a data layer is composed of heterogeneous objects, it may be possible to split it into two layers of large and small objects and use the relevant strategy for each sub-layer. In order to improve the architecture of the system (servers and clients), it is pertinent to use the same grid structure for both vector tiling and spatial indexing (a quad-tree). In that way, the same client cache structure for tiles and vector objects may be used for both strategies.

4.3 LoD-Based Data Extraction

Multi-scale data produced by generalization allow relevant data to be extracted according to the zoom level. It is necessary to synchronize the zoom level with the relevant level of detail (LoD) so that simplified enough data are transferred from the server to the client. Pre-computed multi-scale data should follow the equal information density law [31]: Whatever the visualization scale, the information density should be constant and remain below a threshold. This threshold is both a legibility and performance threshold: It ensures the map is simple enough to be legible, and, in a web context, it also ensures there is no performance issue according to the system capacities (bandwidth, memory, data processing and rendering) – Generalization should be used to ensure vector tiles size is low enough to be transferred and rendered by the client in a satisfying time.

Nowadays, only few simplistic generalization techniques are used in web mapping. In [6], we propose an architecture to improve the use of existing generalization techniques in web mapping. In this architecture, the generalization is shared between the server and the client:

- Multi-scale vector data are computed and stored on the server using model generalization. Model generalization (also called conceptual or semantic generalization) allows a level of detail reduction of the data. Object representing detailed concept are usually aggregated into objects representing more generalized concepts (See figure 5). The geometric level of detail is also reduced according to a target resolution of the data.

- Graphic generalization is performed on the fly and progressively on the client while loading and rendering the vector data. Graphic generalization transforms the map objects to ensure legibility constraints are satisfied. For example, too small objects are enlarged, and too close objects are deformed or displaced (See figure 6). Opening web mapping to vector data makes possible the development of clients with graphic generalization capabilities as described in 32.

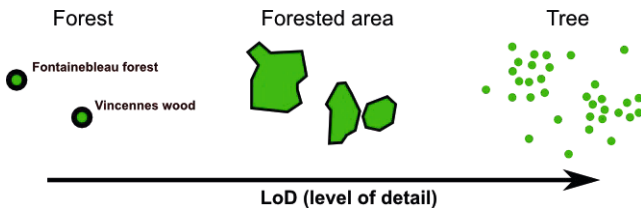


Fig. 5. Model generalization: Forests, forested areas, and trees. Three concepts representing the same reality for different semantic levels of details.

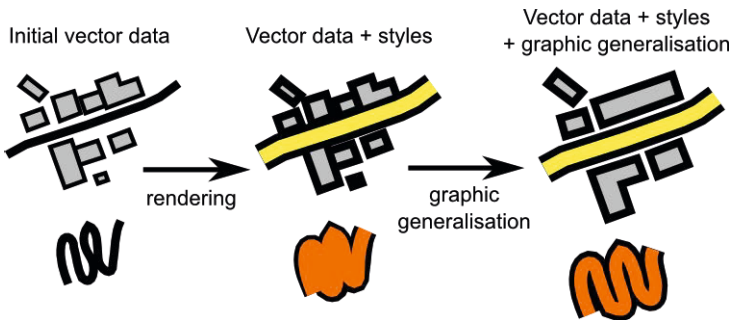


Fig. 6. Graphic generalization (Figure from 6)

5 Experiments

The presented framework has been implemented as part of the OpenCarto project 33. This project aims at providing a software platform to expose advanced spatial data visualisation techniques on the web using vector data. It includes various modules for spatial data import, a component for multi-scale mapping composed of a generic multi-scale data model and generalisation algorithms, some components for vector tiling and spatial indexing, and a vector web mapping client as described in 32.

The prototype has been tested on two kinds of datasets (See figure 7): A dense dataset of small objects (world airports represented as points), and a dataset of large objects (relief contour lines). For both datasets, one generalised data layer has been produced for each zoom layer – the standard mercator zoom levels from

4 to 15 have been tested. The small objects dataset has been generalised using clustering, displacement and filtering algorithms (See figure 7 left); The large objects dataset has been generalised using selection based on contour interval and filtering algorithms (See figure 7 right). These datasets have then been transformed into a hierarchy of 256*256px GeoJSON vector tiles and published using the *http://myurl.org/z/x/y.json* standardised URLs pattern. No spatial indexing service has been tested yet.

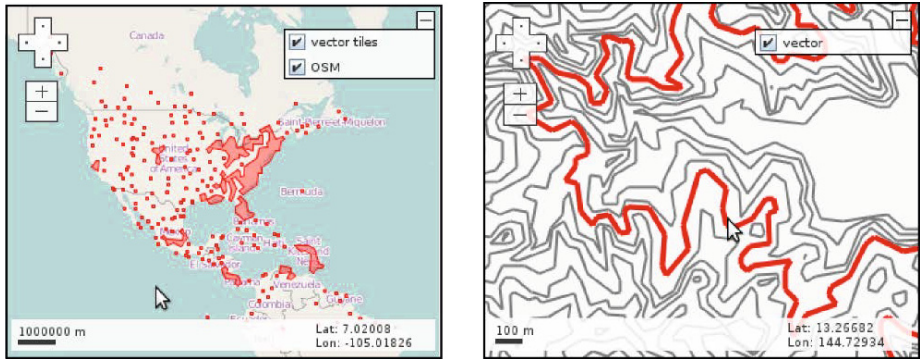


Fig. 7. Test case: Airport point data (left) and relief data as contour lines (right)

At the end of the tile preparation process, the size of the tile repository is 34MB for dataset 1 and 22MB for dataset 2. Without generalisation, these sizes are respectively 242MB and 1.16GB. Without generalisation, the tile size distribution is rather heterogeneous and some tiles have a size of 101MB for dataset 2. With generalisation, the tile size distribution is homogeneous and do not exceed 110KB – this maximum size can be controlled by the generalisation level. For a basic 'spatial exploration' from one point to another and from one zoom level to another, the performance can be measured by the data amount to transit from the server to the client: Because the screen size do not change, the number of vector tiles requested do not change, and because the tile size is low thanks to generalisation, the performance is significantly improved. The spatial exploration is smooth whatever the location and zoom level. No performance issue

Table 1. Comparison raster – vector

Raster	Vector
Resolution	Level of detail
Image pyramid	Multi-scale database
Resampling	Generalization
Raster tiling	Vector tiling / spatial indexing
Raster progressive transmission	Vector progressive transmission

is encountered anymore, mainly thanks to the integrated use of the techniques presented in section 4.

6 Discussion and Conclusion

In this article, the potential benefits and challenges of vector web mapping have been presented. A framework integrating some existing techniques (vector tiling, spatial indexing, multi scale data and generalization) has been proposed, implemented and tested.

A future challenge would be to improve the integration of vector and raster web mapping techniques. Table 1 proposes analogies between raster and vector techniques. Unified data structures and services may be designed to progressively erase the boundary between vector and raster approaches. In a same way that the feature and coverage views can be integrated in GIS [34], vector and raster approaches may be merged in web mapping. A phenomena that appears as objects at some scales and as coverages at other scales may be represented using either raster or vector web mapping services depending on the zoom level.

Furthermore, in order to support the development of vector web mapping and ensure a minimal interoperability between vector servers and clients, specific standards may be proposed. Open formats for vector data and associated style formats are required. The requirements for such format according to the proposed framework would be:

- To allow thin representations of geometry and attributes. The JSON grammar used in GeoJSON format is certainly a pertinent candidate. Standard file compressions may also be used.
- To allow a separation between geometrical and attribute data. This requirement may improve vector tiling performance.
- To allow a separation between object and style description. This separation would enable the reuse of on-line data with personalized styles and, in the same way, the reuse of on-line styles on other data. It would make possible the development of vector style servers, beside vector data servers.
- To allow the definition of dynamic styling behaviors. The way an object displays should not be static - it should depend on its context.
- To allow the definition of object behaviors according different interface events.

A second standardization field may be protocols for client/server communication. Most of the existing international standards (like the ISO and OGC standards WMS, WFS, GML and SLD) have been designed mainly for download services and do not take into account the specific requirements of vector web mapping. Specific services, such as the *Complex Vector Web Service Protocol*, may emerge. The spatial indexing service we have proposed may also be subject to standardization – it may be designed as an extension of WFS.

Furthermore, it may be useful to improve the way the LoD/scale dimension is handled in existing vector data formats. In the same way geographical objects have a spatial and temporal extent, they also have a “scale extent” as formalized

in [30]. This scale extent is a scale range for which the spatial object exists. It would make easier the publication of vector objects on the Internet and their use by vector web mapping applications. The definition of scale extents for vector object exist in KML (with the “lod” element), in SLD (for layers), and also in SVG [35]. Structures and formats to represent multi-scale objects would also be required. For the same reason that there are coordinate reference systems for space, it may be needed to define *scale reference systems*. Such scale reference system would define which zoom levels are supported by the multi-scale vector database – it may be continuous (an object scale extent would be a scale interval) or discrete (an object scale extent would be a set of zoom levels).

Finally, taking into account the high diversity of geographical data available on the Internet, it is necessary to provide generic model and graphic generalization patterns to be adaptable to a wide set of geographical objects. Generic model generalization patterns such as heat maps, cluster hierarchies, multi-scale networks and multi-scale contour maps may be developed in the future.

References

1. Peterson, M.P. (ed.): International Perspectives on Maps and the Internet. Lecture Notes in Geoinformation and Cartography. Springer (2008)
2. Esri: Comparing Vector and Raster Mapping for Internet Applications, An ESRI White Paper. Technical report, ESRI (August 2006)
3. Antoniou, V., Morley, J.: Web Mapping and WebGIS: do we actually need to use SVG? In: SVG Open 2008: 6th International Conference on Scalable Vector Graphics (August 2008)
4. Perron, J.: The future of web mapping (November 2009), <http://www.nsimtech.com/the-future-of-web-mapping/>
5. Ghadirian, P., Bishop, I.D.: Integration of augmented reality and GIS: A new approach to realistic landscape visualisation. Landscape and Urban Planning 86(3-4), 226–232 (2008)
6. Gaffuri, J.: Improving Web Mapping with Generalization. Cartographica: The International Journal for Geographic Information and Geovisualization 46(2), 83–91 (2011)
7. Zhang, Q., Harrie, L.: Placing Text and Icon Labels Simultaneously: A Real-Time Method. Cartography and Geographic Information Science 33(1), 53–64 (2006)
8. Christophe, S.: Creative Colours Specification Based on Knowledge (COLor-LEGend system). The Cartographic Journal, 138–145 (May 2011)
9. Bucher, B., Buard, E., Jolivet, L., Ruas, A.: The Need for Web Legend Services. In: Ware, J.M., Taylor, G.E. (eds.) W2GIS 2007. LNCS, vol. 4857, pp. 44–60. Springer, Heidelberg (2007)
10. Goodchild, M.F.: Citizens as sensors: the world of volunteered geography. GeoJournal 69(4), 211–221 (2007)
11. Craglia, M., de Bie, K., Jackson, D., Pesaresi, M., Remetey-Fülöpp, G., Wang, C., Annoni, A., Bian, L., Campbell, F., Ehlers, M., van Genderen, J., Goodchild, M., Guo, H., Lewis, A., Simpson, R., Skidmore, A., Woodgate, P.: Digital Earth 2020: towards the vision for the next decade. International Journal of Digital Earth 5(1), 4–21 (2011)

12. Harrie, L., Sarjakoski, T., Lehto, L.: A variable-scale map for small-display cartography. In: Joint International Symposium on GeoSpatial Theory, Processing and Applications, ISPRS/Commission IV, SDH 2002, Ottawa, Canada (July 2002)
13. Badard, T., Richard, D.: Using XML for the exchange of updating information between geographical information systems. *Computers, Environment and Urban Systems* 25(1), 17–31 (2001)
14. Spanaki, M., Antoniou, B., Tsoulos, L.: Web Mapping and XML Technologies: A Close Relationship. In: 7th AGILE Conference on Geographic Information Science (April 2004)
15. Antoniou, V., Morley, J., Haklay, M(M.): Tiled Vectors: A Method for Vector Transmission over the Web. In: Carswell, J.D., Fotheringham, A.S., McArdle, G. (eds.) *W2GIS 2009*. LNCS, vol. 5886, pp. 56–71. Springer, Heidelberg (2009)
16. Langfeld, D., Kunze, R., Vornberger, O.: SVG Web Mapping. Four-dimensional visualization of time- and geobased data. In: *SVGOpen 2008* (2008)
17. On Building Pyramid of Geographic Vector Data for OpenStreetMap. In: The XII congress of the International Society for Photogrammetry and Remote Sensing (ISPRS) (August 2012)
18. Jones, C.B., Mark Ware, J.: Map generalization in the Web age. *International Journal of Geographical Information Science* 19(8-9), 859–870 (2005)
19. Cecconi, A., Galanda, M.: Adaptive Zooming in Web Cartography. *Computer Graphics Forum* 21(4), 787–799 (2002)
20. Mackaness, W.A., Ruas, A., Sarjakoski, L.T. (eds.): *Generalisation of Geographic Information, Cartographic Modelling and Applications*. International Cartographic Association Series. Elsevier B.V., Oxford (2007)
21. Ica: Website of the ICA commission on generalisation and multiple representation, <http://generalisation.icaci.org/>
22. Rauschenbach, U., Schumann, H.: Demand-driven image transmission with levels of detail and regions of interest. *Computers and Graphics*, 857–866 (December 1999)
23. Bertolotto, M., Egenhofer, M.J.: Progressive Transmission of Vector Map Data over the World Wide Web. *Geoinformatica* 5(4), 345–373 (2001)
24. Liu, H., Yang, W., Yu, S., Chen, Y.: Progressive Vector Data Transmission Based on Overall Effects. In: *NCM 2008: Proceedings of the 2008 Fourth International Conference on Networked Computing and Advanced Information Management*, pp. 604–608. IEEE Computer Society, Washington, DC (2008)
25. Yang, B., Weibel, R.: Editorial: Some thoughts on progressive transmission of spatial datasets in the web environment. *Computers & Geosciences* 35(11), 2175–2176 (2009)
26. Yang, B., Li, Q.: Efficient compression of vector data map based on a clustering model. *Geo-Spatial Information Science* 12(1), 13–17 (2009)
27. Ai, B., Ai, T., Tang, X.: Progressive transmission of vector map on the web. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 37(pt. B2) (2008)
28. Corcoran, P., Mooney, P., Bertolotto, M., Winstanley, A.: View- and Scale-Based Progressive Transmission of Vector Data. In: Murgante, B., Gervasi, O., Iglesias, A., Taniar, D., Apduhan, B.O. (eds.) *ICCSA 2011, Part II*. LNCS, vol. 6783, pp. 51–62. Springer, Heidelberg (2011)
29. Ramos, J.A., Esperanca, C., Clua, E.W.: A progressive vector map browser for the web. *Journal of the Brazilian Computer Society* 15, 35–48 (2009)

30. van Oosterom, P., Stoter, J.: 5D Data Modelling: Full Integration of 2D/3D Space, Time and Scale Dimensions. In: Fabrikant, S.I., Reichenbacher, T., van Kreveld, M., Schlieder, C. (eds.) GIScience 2010. LNCS, vol. 6292, pp. 310–324. Springer, Heidelberg (2010)
31. Töpfer, F., Pillewitzer, W.: The principles of selection. *Cartographic Journal* 3, 10–16 (1966)
32. Gaffuri, J.: Generalisation on the web: Towards 'scale-aware' web mapping clients. In: Harrie, L., Sandgren, U. (eds.) Workshop on Web Cartography. Lantmäteriet & Lund University (May 2011)
33. OpenCarto project, <http://www.opencarto.goldzoneweb.info/>
34. Cova, T.J., Goodchild, M.F.: Extending geographical representation to include fields of spatial objects. *International Journal of Geographical Information Science*, 509–532 (September 2002)
35. Campin, B.: Use of vector and raster tiles for middle-size Scalable Vector Graphics' mapping applications. In: SVGOpen 2005 (August 2005)

spatial@linkedscience – Exploring the Research Field of GIScience with Linked Data

Carsten Keßler¹, Krzysztof Janowicz², and Tomi Kauppinen¹

¹ Institute for Geoinformatics, University of Münster, Germany
{carsten.kessler,tomi.kauppinen}@uni-muenster.de

² Department of Geography, University of California, Santa Barbara, USA
jano@geog.ucsb.edu

Abstract. Metadata for scientific publications contain various explicit and implicit spatio-temporal references. Data on conference locations as well as author and editor affiliations – both changing over time – enable insights into the geographic distribution of scientific fields and particular specializations. At the same time, these *byproducts* of scientific bibliographies offer a great opportunity to integrate data across different bibliographies to get a more complete picture of a domain. In this paper, we demonstrate how the Linked Data paradigm can assist in enriching and integrating such collections. Starting from the bibliographies of the GIScience, COSIT, ACM GIS, and AGILE conference series, we show how to convert the data to Linked Data and integrate the previously separate datasets. We focus on the spatio-temporal aspects and discuss how they help in matching and disambiguating entities such as authors or universities. We introduce a novel user interface to explore the integrated dataset, demonstrating the potential of Linked Data for innovative applications using spatio-temporal information, and discuss how more complex queries can be addressed. While we focus on bibliographies, the presented work is part of the broader vision of a Linked Science infrastructure for e-Science.

1 Introduction

Over the last decades several conference series and journals have been established to communicate and publish research on Geographic Information Systems and Science. However, without being a member of the research community and its different subfields, it is difficult to judge how the conferences and journals differ, which one should be selected for a specific publication, and where to meet scholars that share related research interests. While calls for papers and editorial boards can be used as indicators, they do not provide enough discriminatory power and overlap to a large degree. A single resource to learn about GIScience-related publications, events, researchers, their interconnections, and research affiliations is missing. While CiteSeer, Google Scholar, DBLP, Springer-Link, or Mendeley offer large amounts of metadata on scientific publications, the links between those datasets that would enable more complex queries are missing. Moreover, even within a specific bibliographic dataset it is difficult to track

the identity of authors and their affiliations. For instance, several manual queries are required to find all spelling variants for a single author. Thus, even simple queries for co-authors that would help journal editors and program chairs to find reviewers without conflicts of interest are virtually impossible. This is especially striking as the information to answer such queries is usually part of the bibliographic data provided by publishers.

In this paper, we demonstrate that the Linked Data paradigm can provide us with the methods to interlink datasets and establish identity by using the spatio-temporal properties for matching and disambiguation. The term *Linked Data* refers to a set of principles to publish machine-readable and understandable data online that has been proposed by Tim Berners-Lee [1]. These principles make use of well-established Web standards for identifying and accessing data sources, along with lightweight semantics to create a global graph of data. This distributed and interlinked collection of datasets is also referred to as the Linked Data Cloud [2] and has been growing rapidly over the past years [3], with some geographic information sources such as GeoNames¹ acting as central hubs.

In this paper, we use an integration scenario for publications from different conference series, i.e., COSIT, GIScience, ACM GIS, and AGILE to illustrate the potential of spatio-temporal properties in Linked Data. As *non-information resources*, researchers or universities can only be in one place at a given time. Spatio-temporal information about these entities can act as identity criteria that allow us to match data that may be registered under different names in different datasets, and disambiguate different events that accidentally share a common (place) name. We discuss how to use the spatio-temporal properties in bibliographic data from conference series proceedings for identity reasoning. Besides the potential of spatio-temporal information to facilitate data integration, we demonstrate how the final product—an interlinked online collection of dereferenceable bibliographic resources, available through a standardized API—can act as the foundation for intuitive, exploratory user interfaces.

We show that by semantically annotating, integrating, and interlinking bibliographic datasets, we can answer complex queries. For instance, due to their history all conferences have their own areas of specialization and, therefore, attract a different audience. Which researchers act as bridges between these communities and conferences, i.e., have published in several conference series? Given a certain sub-field of GIScience, which event offers the highest probability to meet researchers who share the same interests? How do topics evolve over time, gain, and lose interest? We have made all presented datasets, APIs, and user interfaces freely available on the Web at <http://spatial.linkedsience.org> and are constantly enriching them. While we focus on bibliographic data here, the presented work has broader implications on e-Science [4] and is part of the vision of a Linked Science [5] infrastructure.

The remainder of the paper is structured as follows. In the next section, we discuss relevant related work and the used technologies. Section 3 describes the data conversion process and data sources in detail. Section 4 presents the data

¹ See <http://geonames.org>

integration and mapping methodology and evaluates it with sample queries. Finally, we present the system architecture and a user interface that makes our inter-linked and enriched dataset available to the GIScience community. We conclude our work by pointing out limitations and directions for future work.

2 Related Work

Linked Data was proposed by Tim Berners-Lee as a practical, data-driven approach towards the vision of the Semantic Web. The approach consists of four principles [1]: (1) use of URIs as names for things; (2) use of HTTP URIs to enable look-up; (3) use of the Resource Description Framework (RDF) and SPARQL Protocol And RDF Query Language (SPARQL) standards for data encoding and querying; and (4) interlinking of datasets to enable discovery. As these principles build on established standards that have proven useful in the “Document Web”, the approach was quickly adopted by the community [3]. The Linked Data Cloud [2] that was built on these principles is constantly growing. As of September 2009, it consists of more than 31 billion RDF triples, which are the basic building blocks of Linked Data [6].

The bibliographic domain has been one of the first fields to embrace Linked Data as a new way to publish bibliographic records online in a machine-readable way. Several hubs in the Linked Data Cloud provide bibliographic collections, especially with an academic focus, such as the ACM library [2] DBLP [3] or CiteSeer [4]. Major libraries such as the British Library or the German National Library already offer Linked Data services. In order to semantically annotate the published datasets, the community has been working on a number of vocabularies to describe their datasets, of which the bibliographic ontology BIBO is most widely used [7]. The Semantic Publishing and Referencing (SPAR) ontologies are a more detailed attempt towards semantic publishing that goes beyond classical metadata [8]. ArnetMiner [5] is an ongoing research effort based on bibliographic data, building a dataset for exploring aspects such as advisor-advisee relationships [9] or social network mining [10].

Unfortunately, GIScience is under-represented in systems such as ArnetMiner. Many relevant conference series and journals are not listed. Analyzing GIScience as research field is not new; for instance, Skupin has applied document spatialization approaches to visualize the domain [11]. Agarwal et al. [12] have used a social graph to study the interconnectedness of the community, while Grossner and Adams [13] used Latent Dirichlet allocation to study research trends by analyzing the full papers of the last ten COSIT conferences.

Linked Data has also recently been explored by several researchers as a new way of publishing spatio-temporal data. Examples include a Linked Data version of OpenStreetMap [14], Ordnance Survey Linked data [6] and data from the

² See <http://thedatahub.org/dataset/rkb-explorerer-acm>

³ See <http://thedatahub.org/dataset/fu-berlin-dblp>

⁴ See <http://thedatahub.org/dataset/rkb-explorerer-citeseer>

⁵ See <http://arnetminer.org/>

⁶ See <http://data.ordnancesurvey.co.uk/>

Sensor Web [15]. The main motivations are twofold: First, Linked Data enables adding light-weight semantic annotations to the data—a task which has bothered geographic information scientists for more than a decade [16]. Second, Linked Data offers a way to expose geographic information to a wider audience that does not know anything about the standards used in spatial information infrastructures. Recent developments such as the GeoSPARQL [17] query language show a strong tendency in the field to open up and make use of generic standards for data exchange that do not only work for geographic information.

In this paper, we merge the efforts from these two domains. We document how we approached this problem and show how the result enables novel user interfaces and interaction paradigms. As such, this paper is one step in realizing the vision of *Linked Science* [5] as an infrastructure for e-Science, an approach to semantically annotate and interconnect scientific resources such as models, data, methods and evaluation metrics [4].

3 Conversion Process

This section describes the conversion process of bibliographic metadata in BibTeX format into an RDF representation.

3.1 Input Data

The input data for our process consist of metadata for the conference series *International Conference on Geographic Information Science* (GIScience), *Conference On Spatial Information Theory* (COSIT), *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (ACM GIS), and *AGILE International Conference on Geographic Information Science* (AGILE). For each series, the metadata for each paper in BibTeX format were used, containing information about authors, year, title, proceedings title, editors, digital object identifier (DOI), pages, publisher, and author affiliations. The metadata sum up to 1256 papers (110 for GIScience, 331 for COSIT, 699 for ACM GIS, and 116 for AGILE) from a total of 36 proceedings volumes (5 for GIScience, 11 for COSIT, 15 for ACM GIS, 5 for AGILE) [8]. Unfortunately, the metadata for the proceedings of some early conferences of the GIScience, ACM GIS and AGILE series were not available. We are constantly adding and enriching new data, e.g., to integrate the SDH, AutoCarto, and Spatial Cognition conference series.

3.2 From BibTeX to RDF

The first step towards a Linked Data version of the series proceedings is the conversion to RDF. For this purpose, a Java converter has been developed that

⁷ See <http://linkedsience.org>

⁸ Note that these numbers only describe our input data; they are not intended to analyze the domain or make any statements about the importance of the different conference series.

iterates through the collection of BibTeX files and generates RDF statements. As mentioned in Section 2, URIs serve as identifiers for Linked Data resources. It is therefore crucial to develop URI conventions as a first step that defines how the URIs for different kinds of resources will be structured. These URI patterns are then filled by certain properties from the input data:

- Paper:
<http://spatial.linkedscience.org/context/paper/doiDOI>
 Example:
<http://spatial.linkedscience.org/context/acmgis/paper/doi10.1145/1653771.1653787>
- Person:
<http://spatial.linkedscience.org/context/person/personMD5-Hash>
 Example:
<http://spatial.linkedscience.org/page/context/person/person4a54e293d2c33b74e2aab49bb5c182b6>
- Affiliation:
<http://spatial.linkedscience.org/context/affiliation/affiliationMD5>
 Example:
<http://spatial.linkedscience.org/page/context/affiliation/affiliationc429ca266aa3fce217af9c8ef1524f9a>

We followed the strategy to re-use any unique identifiers that were already present in the data, such as DOIs for the single papers, and only created our own identifiers when necessary. In case of the author names and affiliations, we created MD5 hashes from the input strings in order to prevent overly long URIs. At the same time, the hashing also removes any special characters such as umlauts from the URIs. The set of resources that is created by following these URI patterns then needs to be interlinked using properties (also referred to as predicates) from RDF vocabularies. The fields in the BibTeX files can be mapped to RDF properties defined in existing and widely used vocabularies. Hence, there was no need to create new vocabularies. We have used properties from six existing vocabularies: Dublin Core (namespace `dc`), Friend Of A Friend (`foaf`), the Bibliographic Ontology (`bibo`), the Ontology for vCards (`vcard`), and the W3C Basic Geo Ontology (`geo`).⁹ Figure 1 gives an overview of how the different resource types are interlinked and annotated with the properties from those vocabularies.

The `geo:lat`, `geo:lon` and `vcard:ADR` properties shown in the figure cannot be created directly from the BibTeX input. They serve to encode the georeferences for all affiliations. To generate them, we queried the affiliation string from each BibTeX file against the Google Geocoding API,¹⁰ which returns both the

⁹ See <http://dublincore.org/documents/dcmi-terms/>, <http://xmlns.com/foaf/spec/>, <http://bibliontology.com/specification>, www.w3.org/2006/vcard/ns, and <http://www.w3.org/2003/01/geo/> for the respective specifications.

¹⁰ See <http://code.google.com/apis/maps/documentation/geocoding/>.

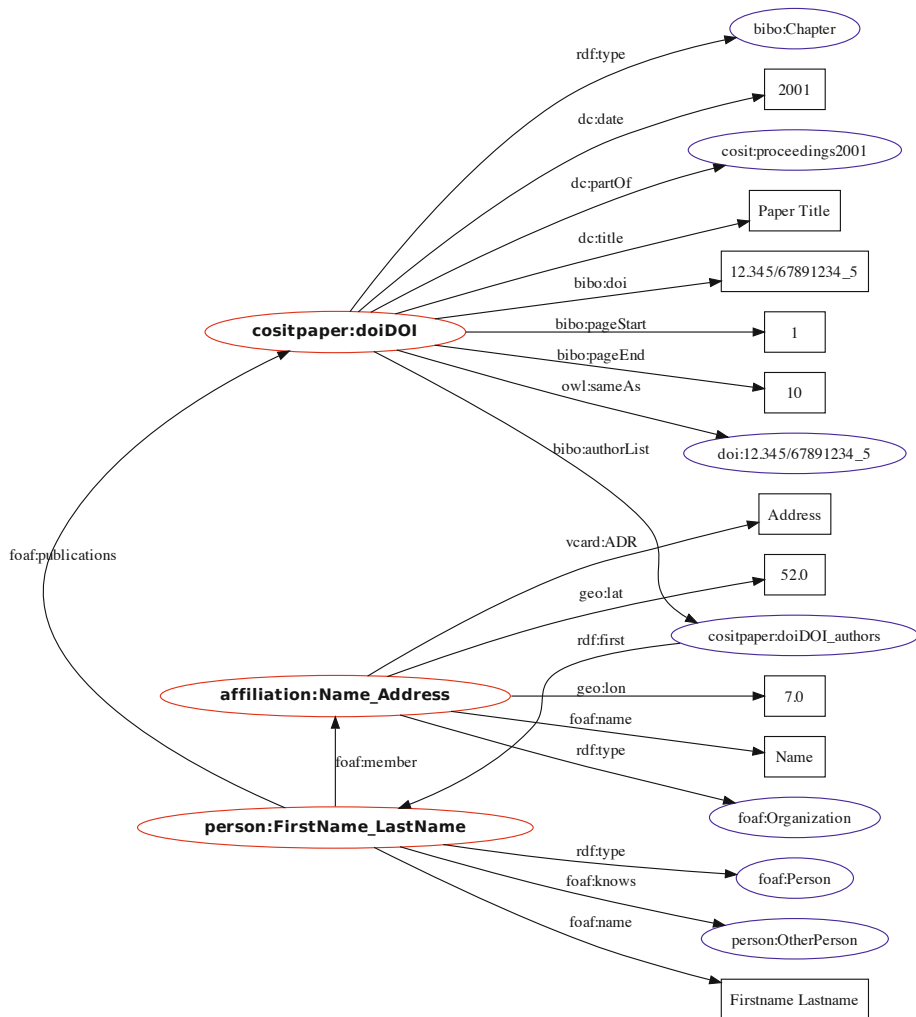


Fig. 1. Overview of interlinking schema. Resources are depicted as ellipses, with the three main types highlighted in bold font. Literals are depicted as boxes.

latitude and longitude for the given location, along with a properly formatted version of the address. The affiliation strings often include very specific information that is not required for the geocoding process, such as working group or department information. This information is not only superfluous for the geocoder, it often even prevents finding a result. To cover these cases, we iteratively removed words from the beginning of the affiliation string until a result was returned, following the rationale that the strings usually start with more specific information and get generic towards the end. With this approach, we could successfully georeference all but 3 of the 1021 distinct affiliations. Note that the total number of affiliations (1021 for 1256 papers) is so high because

we still have separate affiliation resources for every name and spelling variant at this point. We will tackle this issue in the following section.

The author lists and affiliations required a more complex transformation in order to fully represent the original information in RDF. In case of the author lists, it is not sufficient to link a paper to its authors via the `foaf:publications` property. This approach drops the order of authors, as any statement about a resource is treated equally. Therefore, when multiple statements of the same kind are available, they will appear in an arbitrary order in the output, thus losing the information who is first author, second author, and so on. The `author` entry from the BibTeX file is therefore transformed into an RDF list that is linked to the paper resource via the `bibo:authorList` property, as shown in Figure 2.

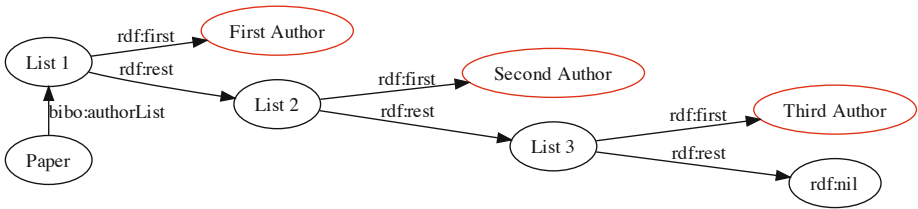


Fig. 2. Schematic view of nested author lists. `rdf:nil` marks the end of the list.

Likewise, it is not sufficient to only attach the affiliations directly to the authors via `affiliation:X foaf:member person:Y`, as this approach would lose the information *when* the author Y was affiliated with the institution X. We therefore reify the affiliation relationship, i.e., we turn the membership property into a resource, that has subject, predicate, and object attached as properties. This allows us to attach additional metadata to the whole statement—in this case, we attach the year of publication to retain the information when this relationship was valid:

```

membership123 rdfs:type          rdfs:Statement ;
               rdfs:subject      affiliation:X ;
               rdfs:predicate    foaf:member ;
               rdfs:object       person:Y ;
               dc:date           "2005" .
  
```

In order to include the social network aspect in the dataset, we also add a `foaf:knows` link between two persons if they have published a paper together.

4 Data Integration and Mapping

This section describes the mapping approach we applied to integrate and enrich the data. In order to keep track of data provenance, the RDF data for each conference is stored in a separate named graph (e.g. <http://spatial.linkedscience.org/context/giscience>). This allows us to keep track of where a specific triple

comes from. In the following, we describe how we consolidated multiple URIs for the same person or affiliation and how we interlinked the data across the four named graphs.

4.1 Mapping Approach

The high number of affiliation resources generated by our initial conversion step – 1021 affiliations from 1256 papers – shows that people use different spelling variations of their affiliation in different papers. The same applies to author names, where some papers include first name and middle initials, others include only the first name (in potential spelling variants, e.g. *James* vs. *Jim*), or just initials. While it is fairly easy to see for a human reader that the author names *Michael F. Goodchild*, *M.F. Goodchild* and *Mike Goodchild* probably refer to the same person, these heterogeneities in the input data create challenges for the RDF generation, where we strive for a single URI that identifies a person or affiliation.

We approached this problem by combining spatial distance measures with string similarity measures. This mapping was carried out with the Silk link discovery framework [18]. In order to consolidate the high number of URIs for affiliations, we have compared the names of all organizations that were not located more than 10km apart. We had to pick such a comparably high range for the spatial search because the output of the geocoding process varied widely in terms of the levels of the location that were recognized. Depending on the input string, the results range from locations of exact street addresses or intersections to cases where only the state or even only the country was recognized. The 10km radius was selected as a rule of thumb in order to compare all places that have been automatically georeferenced into the same city. Within this range, all organization names were compared based on the Jaccard similarity coefficient [19]. It provides a normalized measure of token-based string similarity, calculating the size of the intersection divided by the size of the union of two sets of words. It ignores the order of the words in the affiliation title, so that e.g., the two strings ‘*University of California NCGIA and Geography Department Santa Barbara*’ and ‘*Department of Geography University of California Santa Barbara*’ still yield a high similarity (0.78). The application of the Jaccard coefficient prevents matching of different organizations that are located near each other.

In case of the author names, we have applied a Levenshtein distance measure as a first step, which counts the number of editing steps between two strings [20]. Two names with an overall edit distance below 4 *and* exactly matching last names were used as candidates for consolidation. To prevent merging two different persons that happen to have very similar names, their affiliations were taken into account: if no common affiliations could be found, these persons were kept separate.

For both the affiliations and authors, the mappings were stored into a separate named graph at <http://spatial.linkedsience.org/context/sameas>. Each triple in this graph links two resources that have been identified as representing the same organization or person using the owl:sameAs property. As this property

formally assigns *all* information about one of the linked resources to both of them, special care must be taken not to map resources if there is any doubt that they really refer to the same person or organization [21].

4.2 Linking Out

The fundamental idea of Linked Data is to interlink resources across different datasets. This process enriches local datasets with external data and embeds a dataset into the global graph. Our conference dataset contains external links to the Semantic Web Service of the GeoNames gazetteer.^[11] For each georeferenced affiliation, we have created an outgoing link to the closest entry in Geo-Names through its `findNearby` API, such as `<http://spatial.linkedscience.org/context/affiliation/affiliationdb445b559df12b0d28fe03b432be04a0> foaf:basedNear <http://sws.geonames.org/2867543/> .`

Adding a pointer to GeoNames to the dataset may seem redundant, since the affiliations are already georeferenced. Using the external data provided by GeoNames, however, enables new spatial queries, especially concerning administrative hierarchies. As every resource in GeoNames is part of a hierarchy tree, these outgoing links enable queries by country or continent, for example. The outgoing links to GeoNames are stored in a separate graph at <http://spatial.linkedscience.org/context/geonames>.

5 Architecture and Interaction

This section describes the client-server architecture of the application built on top of the dataset. It shows the user interface, explains the interaction workflow, and shows how the SPARQL endpoint can be used for complex queries.

5.1 Architecture

Having an integrated and inter-linked data set of GIScience-related conference series allows us to develop novel applications on top of it. The Web application described in this section is available at <http://spatial.linkedscience.org>. It is based on a client-server architecture that uses asynchronous JavaScript requests (AJAX) for communication. Figure 3 shows an overview of the architecture: The server hosts a static HTML interface, served through an nginx HTTP server; and a Parliament triple store that hosts the data and offers a GeoSPARQL endpoint for querying the data.^[12] With this setup, it is possible to deliver an empty visualization frame to the client, which then updates the shown data after every user interaction by fetching the corresponding data from the SPARQL endpoint, without interrupting the user workflow by reloading the whole page.

¹¹ See <http://www.geonames.org/ontology/>

¹² See <http://nginx.org/en/> and <http://parliament.semwebcentral.org/>. Both components are free and open source software.

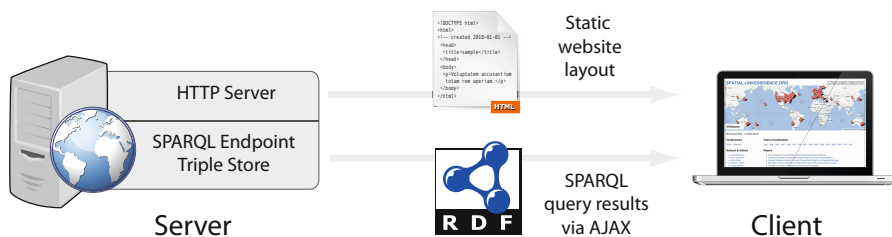


Fig. 3. Architecture overview

5.2 User Interface and Interaction Workflow

The user interface shown in Figure 4 consists of five parts that show different facets of the dataset: the map for affiliations, a list of the conference series, a list of the years of publication, a list of all authors in the dataset, and a list of the titles of all papers. Each of the shown resources can be clicked to get further information. Clicking a marker on the map, for example, lists all authors affiliated with the corresponding organization, along with their papers, the years of publication, and conference series where they have published. Likewise, clicking a year will list all papers that have been published that year, along with their authors, affiliations, and the conferences that took place that year. With all data acting as thematic filters to the dataset, the user interface offers an *exploratory* interaction approach that allows the user to easily browse and navigate the collection.

Figure 5 shows the map visualization that is triggered by clicking an author name. In this case, all affiliations of this author are selected from the dataset, ordered by date and connected by a polyline on the map. This visualization requires the reification discussed in Section 3.2, which allows us to annotate the `foaf:member` property with a timestamp.

Any click on an element of the user interface causes an AJAX query to the server, which returns the corresponding results from the SPARQL endpoint in Javascript Object Notation (JSON) and visualizes them on the map. This asynchronous client-server interaction is handled by the JQuery framework.¹³

5.3 Complex Queries

While the previous subsection discussed querying via the graphical user interface, more complex queries can be directed to the SPARQL endpoint. For instance, to study the differences between sub-communities and their preferred conferences, one may query for those researchers who have published at all major series and, thus, act as bridge builders. The following query selects authors that have papers at ACM GIS, COSIT, and GIScience.

¹³ See <http://jquery.com/>



Fig. 4. Screen shot of the user interface of <http://spatial.linkedscience.org>



Fig. 5. Example of an author trace for Matt Duckham, showing the author’s different affiliations including time stamps

```
prefix foaf: <http://xmlns.com/foaf/0.1/>
SELECT DISTINCT ?author ?name WHERE {
  GRAPH <http://spatial.linkedscience.org/context/acmgis> {
    ?author foaf:publications ?a ; foaf:name ?name .}
  GRAPH <http://spatial.linkedscience.org/context/cosit> {
    ?author foaf:publications ?c .}
  GRAPH <http://spatial.linkedscience.org/context/giscience> {
    ?author foaf:publications ?d .}
}
```

Table 1. Authors that published full papers at ACM GIS, GIScience, and COSIT. Authors marked by an (^a) also published full papers at the AGILE series.

1. Benjamin Adams	7. Mark Gahegan	13. Andrea Rodríguez
2. Christophe Claramunt (^a)	8. Krzysztof Janowicz (^a)	14. John Stell
3. Matt Duckham	9. Christopher B. Jones	15. Egemen Tanin
4. Max J. Egenhofer	10. Lars Kulik	16. Stephan Winter (^a)
5. Leila De Floriani	11. Kai-Florian Richter	17. Michael Worboys
6. Andrew U. Frank (^a)	12. Claus Rinner	

Out of the resulting 23 researchers, we have manually selected those that have full papers in all series; see Table 1. Note that during the last 20 years, some of the conferences have changed their paper categories. We therefore manually excluded extended abstracts from ACM GIS. To show how the list of researchers changes if we add another conference series, we have additionally filtered for researchers that have published full papers at AGILE. While this is an international conference series, it is organized by the Association of Geographic Information Laboratories for Europe and, thus, rather attracts researchers that are or have been based in Europe.¹⁴

A majority of the authors listed in Table 1 have either a background in computer science or are working together with computer scientists. This is due to the strong focus of ACM GIS on *'algorithmic, geometric, and visual considerations'*¹⁵ and a noticeable difference to the GIScience conference series. Adding AutoCarto and the Spatial Cognition series to the query would change this picture and highlight different researchers and associated topics.

6 Conclusions

Collections of bibliographic metadata allow for a detailed analysis of a research field and the corresponding community. So far, publishers and libraries as operators of such collections have come short of tapping this potential. In this paper, we have described a conversion and enrichment process based on the Linked Data paradigm that demonstrates the potential of such collections for the field of Geographic Information Science. We have discussed how the conversion and interlinking processes have been implemented. This workflow transforms the input into RDF and makes use of different online APIs and existing Linked Data sources for data consolidation and enrichment. We have shown how the spatio-temporal properties in the data can be exploited for more efficient data integration and reconciliation of resources from different origins.

Since we use the common BibTeX format for the input data, the collection can be easily extended with further proceedings and additional conference series, which will be the next step to make <http://spatial.linkedscience.org>

¹⁴ So far, our data only covers AGILE papers published by Springer (starting 2007).

¹⁵ See <http://www.sigspatial.org>.

a reference portal for the community. Moreover, we plan to add new functionality such as free-text search. Adding more data will also bring up new research challenges, as an increase in the number of publications and authors brings up new challenges for the organization of the data. In order to facilitate browsing the collection by topic, we plan to add further keywords to the publications. Previous research has shown that such keywords can be extracted using Latent Dirichlet allocation [13]. These annotations would facilitate new kinds of analysis on the contents, such as the development of a certain research topic over time. While more data is important, mining for more relations is of equal value. In the future, we plan to subclass our *knows* relation with a supervisor-student relations and add thematic roles such as reviewers or organizers. In order to generalize this approach and make it useful for other research fields, some analyses would have to be reconsidered. The convention of the first author being the main investigator on the paper, for example, is not consistent across all fields.

Acknowledgements. This research has been partly funded by the *Linked Open Data University of Münster* project (see <http://lodum.de>), the International Research Training Group *Semantic Integration of Geospatial Information* (DFG GRK 1498), and the 52° North semantics community.

References

1. Berners-Lee, T.: Linked Data – Design Issues (2009), <http://www.w3.org/DesignIssues/LinkedData.html> (last accessed December 29, 2011)
2. Cyganiak, R., Jentzsch, A.: Linking Open Data cloud diagram (2011), <http://richard.cyganiak.de/2007/10/lod/> (last accessed January 06, 2012)
3. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data – The Story So Far. *International Journal on Semantic Web and Information Systems* 5(3), 1–22 (2009)
4. Brodaric, B., Gahegan, M.: Ontology use for semantic e-science. *Semantic Web* 1, 149–153 (2010)
5. Kauppinen, T., Baglatzi, A., Keßler, C.: Linked Science: Interconnecting Scientific Assets. In: Critchlow, T. (ed.) *Data Intensive Science*. CRC Press, USA (2012)
6. Bizer, C., Jentzsch, A., Cyganiak, R.: State of the LOD Cloud (2011), <http://www4.wiiss.fu-berlin.de/lodcloud/state/> (last accessed December 29, 2011)
7. Giasson, F., D’Arcus, B.: Bibliographic Ontology Specification (2009), <http://bibliontology.com/specification> (last accessed December 29, 2011)
8. Shotton, D., Portwin, K., Klyne, G., Miles, A.: Adventures in Semantic Publishing: Exemplar Semantic Enhancements of a Research Article. *PLoS Comput. Biol.* 5(4) (2009)
9. Wang, C., Han, J., Jia, Y., Tang, J., Zhang, D., Yu, Y., Guo, J.: Mining advisor-advisee relationships from research publication networks. In: *Proceedings of the 16th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, pp. 203–212. ACM (2010)
10. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: Extraction and mining of academic social networks. In: *Proceedings of the 14th ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*, pp. 990–998. ACM (2008)

11. Skupin, A., Ligigio, R.: Label determination for document spatialization. In: International Conference on Geographic Information Science (GIScience 2006), Münster, Germany, September 21-23 (2006)
12. Agarwal, P., Béra, R., Claramunt, C.: A Social and Spatial Network Approach to the Investigation of Research Communities over the World Wide Web. In: Raubal, M., Miller, H.J., Frank, A.U., Goodchild, M.F. (eds.) GIScience 2006. LNCS, vol. 4197, pp. 1–17. Springer, Heidelberg (2006)
13. Grossner, K., Adams, B.: COSIT at 20: Measuring Research Trends and Interdisciplinarity. In: Conference on Spatial Information Theory (COSIT 2011) (2011)
14. Auer, S., Lehmann, J., Hellmann, S.: LinkedGeoData: Adding a Spatial Dimension to the Web of Data. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 731–746. Springer, Heidelberg (2009)
15. Kefler, C., Janowicz, K.: Linking Sensor Data – Why, to What, and How? In: Taylor, K., Ayyagari, A., De Roure, D. (eds.) Proceedings of the 3rd Int. Workshop on Semantic Sensor Networks 2010 (SSN 2010) at 9th Int. Semantic Web Conference (ISWC 2010), Shanghai, China. CEUR-WS, vol. 668 (2010)
16. Bishr, Y.: Overcoming the Semantic and Other Barriers to GIS Interoperability. *International Journal of Geographic Information Science* 12(4), 299–314 (1998)
17. Open Geospatial Consortium: OGC GeoSPARQL – A Geographic Query Language for RDF Data (2011), Request for comments available from <http://www.opengeospatial.org/standards/requests/80> (last accessed December 29, 2011)
18. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Discovering and Maintaining Links on the Web of Data. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 650–665. Springer, Heidelberg (2009)
19. Tan, P.N., Steinbach, M., Kumar, V.: *Introduction to Data Mining*. Addison Wesley (June 2006)
20. Levenshtein, V.: Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady* 10(8), 707–710 (1966)
21. Halpin, H., Hayes, P.J.: When owl:sameAs isn't the Same: An Analysis of Identity Links on the Semantic Web. In: Workshop on Linked Data on the Web (2010)

Crowdsourcing Satellite Imagery Analysis: Study of Parallel and Iterative Models

Nicolas Maisonneuve and Bastien Chopard

Computer Science Department, University of Geneva, Switzerland

Abstract. In this paper we investigate how a crowdsourcing approach i.e. the involvement of non-experts, could support the effort of experts to analyze satellite imagery e.g. geo-referencing objects. An underlying challenge in crowdsourcing and especially volunteered geographical information (VGI) is the strategy used to allocate the volunteers in order to optimize a set of criteria, especially the quality of data. We study two main strategies of organization: the parallel and iterative models. In the parallel model, a set of volunteers performs independently the same task and an aggregation function is used to generate a collective output. In the iterative model, a chain of volunteers improves the work of previous workers. We first study their qualitative differences. We then introduce the use of Mechanical Turk Service as a simulator in VGI to benchmark both models. We ask volunteers to identify buildings on three maps and investigate the relationship between the amount of non-trained volunteers and the accuracy and consistency of the result. For the parallel model we propose a new clustering algorithm called *democratic clustering algorithm* DCA taking into account spatial and democratic constraints to form clusters. While both strategies are sensitive to their parameters and implementations we find that parallel model tends to reduce type I errors (less false identification) by filtering only consensual results, while the iterative model tends to reduce type II errors (better completeness) and outperforms the parallel model for difficult/complex areas thanks to knowledge accumulation. However in terms of consistency the parallel model is better than the iterative one. Secondly, the Linus' law studied for OpenStreetMap [7] (iterative model) is of limited validity for the parallel model: after a given threshold, adding more volunteers does not change the consensual output. As side analysis, we also investigate the use of the spatial inter-agreement as indicator of the intrinsic difficulty to analyse an area.

Keywords: Volunteer Geographical Information, crowdsourcing, satellite image analysis.

1 Introduction

With the emergence of web 2.0 practices, new participatory approaches in GIS emerged in the last decade. In this paper we investigate the potential of crowdsourcing for the analysis of satellite imagery. Crowdsourcing can be described

as a method of distributed problem-solving to non experts. By definition crowdsourcing represents the act of outsourcing a task performed by employees to an undefined, generally large network of people or community in the form of an open call [8]. Some commonly known examples include Wikipedia and Open Street Map for GIS initiatives. What mechanism at the individual and collective levels will enforce the accuracy and consistency of results? How many volunteers are needed for a given level of quality? How to avoid to waste the time of volunteers while consolidating the quality?

We conducted a set of experiments to test two common organizations enforcing quality: parallel and iterative. In a parallel model, a set of volunteers performs independently the same task and an aggregation function is used to generate a collective output. In an iterative model, a chain of volunteers is used to iteratively improve the work of previous workers (Wikipedia's style).

To focus our study, we did not consider issues related to the division and distribution of a large area to analyze among the volunteers. We also did not consider training methods to improve the quality. Rather we only focused on the nature of the organization and its impact on the quality, assuming that the tasks do not need special training. We asked volunteers to identify buildings on a set of maps. This task could be part of a larger effort to support experts in the analysis of satellite imagery for damage assessment after a humanitarian crisis.

The paper is organized as follows. In section 2 related works in the field of VGI are described. In section 3 we start discussing about the qualitative differences between the parallel and iterative strategies. Then, we explain, in section 4, the experimental setup used to test both models and introduce the use of the *Mechanical Turk Service* as a simulator in VGI to assess them. We present the experiment and its results in section 5 for the parallel model and in section 6 for the iterative model. We finally summarise the findings in the conclusion 7.

2 Related Work in VGI

Concerning the parallel model, in 2000 the NASA launched a first experimental project to assess the reliability of the public in the identification of craters on Mars [9] using a parallel model. But the study did not mention type I errors (fake alarm) and did not explicitly characterize the relationship between the number of volunteers and the accuracy of the results.

More recently such approach was applied in a search and rescue context, to identify missing individuals: the scientist Jim Gray [6] and the aviator Steve Fossett [5]. The method was to divide satellite imagery in small squares, each issued to n people. None of these initiatives managed to find them. In the case of Fossett only 2/3 of the jobs (on 3 millions of tasks) were completed due to the high level of redundancy required (10 reviews per square), raising the question of the optimality of such allocation.

An inverse situation occurred for damage assessment in recent crisis (e.g. Haiti) [17]. Official institutions requested the help of volunteers to speed up damage assessment. For Haiti, a total area of 346 km² was assessed in 96h after

the earthquake (30,000 buildings). While the response was very fast, one problem was the poor reliability of the output due to a lack of a redundancy mechanism (1 volunteer per area), requiring the review of the whole work by experts.

On the other hand, the iterative model is the common process to incrementally improve the quality of user-generated content used on web 2.0 platforms (Wikipedia or Open Street Map). Recently [7] showed the validity of the Linus' Law [16] "given enough eyeballs, all bugs are shallow" used in software development for open source projects to Open Street map by showing the correlation between the number of contributors and the completeness of an area.

Both parallel and iterative models have been used in different contexts, we propose here to study them qualitatively and quantitatively through experiments.

3 Organizational Models of Collective Work

Our approach is built upon the stream of two research domains: the emerging field of human computation [15] which is a paradigm for utilizing human processing power to solve problems that computers cannot solve, and the domain of collective problem solving and organizational studies. We consider two models to process information: the parallel and iterative models. In the parallel model, n volunteers perform independently the same task generating n individual solutions. An aggregation function is used to produce a collective output (see Fig. 1 (a)). In an iterative model, a chain of volunteers is used to review and improve an unique solution sequentially (Fig. 1 (b)).

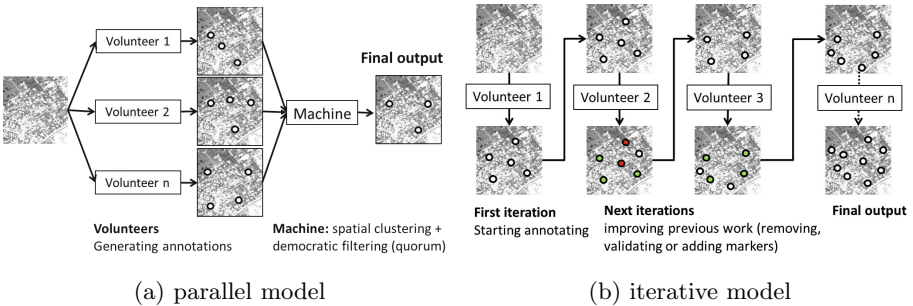


Fig. 1. Schema of the parallel and iterative models

3.1 Parallel vs. Iterative Model

In this section we characterize a set of differences between the parallel and iterative models.

Problem Divisibility. In the parallel model each participant solves the problem independently and thus alone. A problem which is too complex to be solved by one person should then be divided in easier pieces, according to a heuristics

that need to be defined. Such a difficulty is less critical in the case of the iteration model. The whole complexity of the problem can be presented at once. One volunteer can start but not complete the problem, and next participants improve the result. Therefore the nature of the problem and its divisibility can restrict the choice of one approach over the other.

Exploration / Exploitation Trade-Off. A common issue on collective problem solving [10,4,13] is the exploration-exploitation trade-off emerging from the structure of the organization. Networked organizations like iterative models can benefit from the experience of others via the diffusion of knowledge. But exploiting previously discovered solutions can lead to a premature convergence on suboptimal solutions. On the other hand, in the parallel model, individuals are unable to copy one another, leading to a broader exploration in the search space and thus generating a greater diversity of solutions.

Mechanism Enforcing Quality. The concept of wisdom of crowd [19] and collective intelligence [22] lies on the empirical evidence that the aggregation of diverse independently-deciding individuals is likely to make certain types of decisions and predictions better than those of a few experts. Thus, an unbiased approach like the parallel model better supports this property than the iterative model. However, the critical question about the aggregation remains to be considered. The iterative model integrates more naturally the notion of improvement, but it is very sensitive to vandalism (e.g. spamming in Wikipedia). Furthermore, as discussed above, the social influence can impact negatively the collective output [11] due to the path dependency effect [2]: once past decisions have become sufficiently informative, later members simply copy those around them.

Task and Effort. In the human computation field, [12] categorizes the nature of the tasks according to two types: generation of information, and the evaluation and selection of information. For the parallel model the human effort (with potentially the task of aggregating annotation) are related to creation tasks, whereas the iterative model also enables the reviewing. Thus, the effort required can be different: starting from scratch to produce an output requires a priori more effort than reviewing or improving a previous result.

The properties of the parallel and iterative models are summarized in Table 1.

4 Experimental Setup

To evaluate each model we choose the problem of identifying buildings within three different areas using a web platform enabling people to annotate them. We chose 3 areas having different topologies and difficulties (cf Fig. 2). Each map was annotated by one expert in the same conditions as volunteers: 183 buildings were found for map 1, 418 for map 2 and 194 for map 3.

Table 1. Characteristic of parallel/Iterative models

	Parallel model	Iterative model
Process	Same task given to n volunteers generating n solutions aggregated into an unique solution.	Chain of n volunteers reviewing and improving a unique solution.
Type of problem	Problem divisible in tasks completable at an individual level + strategy to merge results	usable for complex problem not easily divisible at an individual level.
Quality mechanism	redundancy + diversity, but useless redundancy for obvious decisions	sequential improvement but path dependency effect + sensitivity to vandalism
Optimisation tradeoff	Independency of decisions emphasizing exploration strategy to solve problem	diffusion of knowledge / copy emphasizing exploitation strategy
Task & effort	generating information from scratch (+ aggregation)	reviewing + improving previous work

4.1 Metrics

To evaluate the performance of a model, we use the traditional recall and precision metrics [1] in the context of VGI to measure type I (false positive) and type II errors (false negative).. In our context, the precision rate P_m is the probability that an annotation is a building for a given model m . The recall rate R_m is the probability that a building is identified. The F-measure F_m is the weighted measure of both the precision and recall rates. Such metrics enable us to measure the trade-off between annotations that are not actual buildings (false positive errors) and not identified buildings (false negative errors). Both metrics are interconnected: mechanisms increasing the precision rate also increase the risk of accepting false positives and thus decrease the recall rate. Mechanisms improving the recall rate increases the risk of accepting false negatives and thus decreases the precision rate.

Let us define $A \cap_\epsilon B$ the intersection of two sets of points, A and B , with a tolerance ϵ as the set of parwises (a,b)

$$A \cap_\epsilon B = \{(a, b) \in A \times B, d(a, b) < \epsilon\}, \quad (1)$$

with $d(a, b)$ the Euclidean distance between point a and b and with a one-to-one matching constraint i.e. $\forall (a_i, b_i)$ and $(a_j, b_j) \in A \cap_\epsilon B, a_i \neq a_j, b_i \neq b_j$.

With this notion, we define the precision, recall and F-measure as

$$P_m = \frac{|V_m \cap_\epsilon E|}{|V_m|}, R_m = \frac{|V_m \cap_\epsilon E|}{|E|}, F_m = 2 \frac{P_m R_m}{P_m + R_m} \quad (2)$$

with E the set of annotations generated by the expert and V_m the set of annotations representing the output generated by volunteers organised according to a given model m (parallel or iterative).

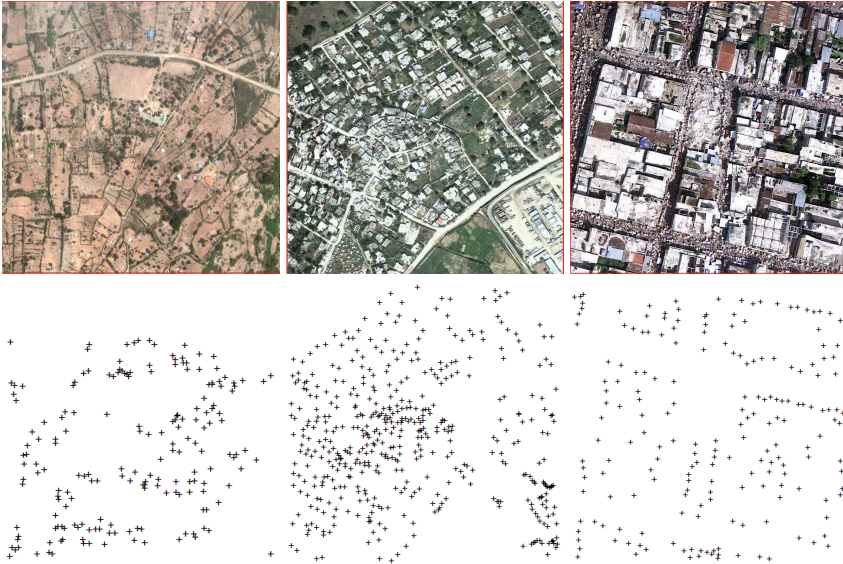


Fig. 2. Three areas with different topologies and densities of buildings. The first row represents the satellite images and the second one the buildings identified by experts (gold standard). From the left, the first column (map 1) represents a sparse area in the Rusinga Island, the 2nd column (map 2) an area in Haiti mixing sparse and dense areas of buildings and the last column (map 3) represents a dense area of Port au Prince in Haiti.

4.2 MechanicalTurk as Simulator in VGI

To recruit participants we used the Amazon Mechanical Turk service (MT). MT is a general-purpose labor market created for crowdsourcing diverse short tasks that are easier to complete for humans than machines (e.g. image labeling, sentiment analysis) in exchange of small financial rewards [18]. Such service became increasingly popular for scientists (e.g. in natural language processing, information retrieval and machine learning) because it demonstrates faster and cheaper web-based experiments with an access to a broader population than students in lab experiments [14]. As far as we know this method has not been experimented to simulate and benchmark organizational models in VGI as we propose here.

5 Experiment 1: Exploring Parallel Model

The goal of this experiment is to understand the relationship between the number of participants and the accuracy of the output using a parallel model for three different maps. The instructions on the on-line platform were to identify and click on every building within a given area.

Since each area was chosen large to gather enough annotations, we cut them in four smaller parts. To realize the annotation task, we present consecutively each part in a random order. The user can not submit a contribution without having reviewed the four small parts. To avoid spammers the system accepts a contribution only if the volunteer has created at least 50 markers (1/4 of the gold standard).

We then tested several aggregations technics. Related crowdsourcing techniques have already been studied to classify a predefined set of images or texts [20,21], asking the opinion of several labelers for each item. In this case, volunteers play the role of (noisy) classifiers. In our case, the map is a continuous space with no predefined distinct set of items. So volunteers play the role of producers of data, extracting all the potential regions of interest (ROI) - buildings -, as well as their classical role of classifiers, filtering only relevant ones at a collective level. Our crowdsourcing method can be decomposed as follows:

1. Generate n tasks for n volunteers and send them to MT.
2. Once the tasks are completed, cluster spatially the annotations to find all the potential ROIs (buildings).
3. Validate or not each ROI according to the level of inter-agreement among the volunteers.

5.1 Clustering Points

Grid-based Clustering. We first tested a coarser method by using a grid-based clustering. For each volunteer V_k in a group of n volunteers, we discretize the space in a grid of $\ell \times m$ cells with $width(cell) = height(cell) = \epsilon$, the tolerance distance. We then assign to each cell the value 1 or 0 according to the presence or absence of annotations. We finally aggregate and validate the results according to an agreement ratio q (see section 5.2) such that each output cells c_{ij}^* is computed as the following:

$$c_{ij}^* = \begin{cases} 1, & \text{if } \frac{1}{n} \sum_{k=1}^n c_{ij}^k \geq q \\ 0, & \text{else} \end{cases} \quad (3)$$

with c_{ij}^k the opinion of volunteer V_k about the presence or absence of buildings on cell (i, j) in the grid. The problem of this approach is the potential unmatching with the topology of the map and its partitioning in a grid, possibly generating quite low performance results (see results in Fig. 6).

Density-based algorithm. Due to the poor performance of the grid-based algorithm, we have tested the *density-based spatial clustering for spatial noise* (DB-SCAN) [3] which requires two parameters: the minimal reachability distance eps and the minimum number q of points required to form a cluster. The parameter q could be seen as a decision threshold in a voting process to validate or reject a cluster of points: if more than q annotations are close enough, we consider it as q volunteers voting for the same area. Once the clusters are defined, we compute their centroids as buildings.

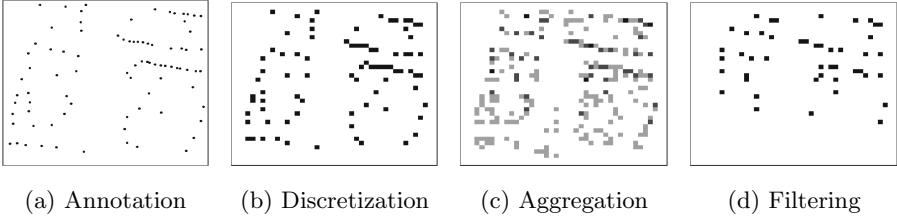
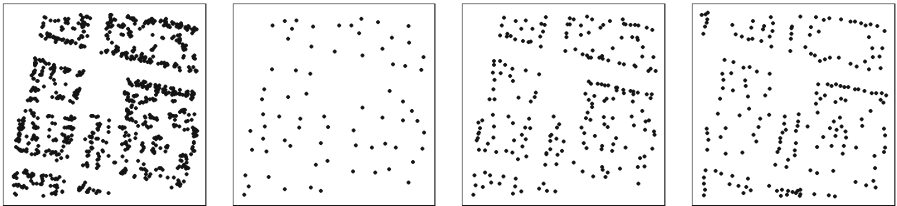


Fig. 3. Grid-based clustering. (3a) represents the annotations of a volunteer for map 3 that are discretized using a grid with 6-meter resolution in (3b). Once discretized they are aggregated with other volunteers, producing the spatial inter-agreement map, see (3c). The darker the cell, the more volunteers made a marker inside it. Finally each cell is validated or not according to a minimum agreement threshold, see (3d).

Democratic clustering algorithm. The main problem with DBSCAN is the lack of a democratic aspect. Firstly several close points selected by the same user are enough to validate a ROI, although the annotations of different volunteers should be required. Secondly, DBSCAN performs worse with a large amount of volunteers in dense images since it is sensitive to the chaining-effect (see Fig. 4).

To tackle this issue, we proposed a clustering algorithm termed *democratic clustering algorithm* (DCA) using spatial and democratic constraints: a cluster should be made of close enough points, originating from different users.

To meet this goal, we changed the notion of spatial neighborhood for a more democratic version. Let A_i be the set of annotations made by the i^{th} volunteer, and A the set of all the annotations from a group g of n volunteers. We introduce the *democratic neighborhood* of a point $a \in A$, as the set $N^g(a)$ of points, defined as $N^g(a) = \{a_1^*, a_2^*, \dots, a_n^*\}$ with $a_i^* = \underset{a_i \in A_i}{\operatorname{argmin}}(\operatorname{dist}(a, a_i))$. In other words, $N^g(a)$ is the set made of the observation a_i from each volunteer i in the group g which is the nearest to a .



(a) Superposition of 10 volunteers (b) DBSCAN results (c) DCA results (d) Gold standard

Fig. 4. DBSCAN vs DCA. The DBSCAN method performs worse with large amount of volunteers in dense images (map 3) due to the chaining effect generated by the noise of more volunteers: 120 clusters found with 3 volunteers while only 71 clusters found with 10 volunteers

The process of the clustering algorithm is the following. Step (1): we take a random point $a \in A$ and build $N^g(a)$. Step (2): we filter $N^g(a)$ according to a maximum distance ϵ to get $N_\epsilon^g(a) = \{a_i^*, d(a_i^*, a) < \epsilon\}$. $N_\epsilon^g(a)$ represents a potential cluster. Step (3): since all volunteers have the same weight, we validate each cluster according to the number of vote i.e. the fraction of volunteers who have an annotation inside $N_\epsilon^g(a)$. We introduce a decision threshold q and cluster a is accepted if $|N_\epsilon^g(a)|/n > q$. Step (4): when a cluster is validated, we define the location of a building as its centroid, and we remove $N_\epsilon^g(a)$ from A . If the cluster is not validated we just remove a from A . Finally we iterate the process from step (1) until A is empty.

5.2 Voting Process and Decision Threshold

A question is now to chose the appropriate decision threshold $q = k/n$ to validate each ROI (cluster), where k is the number of volunteers having voted for it, and n is the number of participants.

To understand the impact of q on the quality of the result, we compute the average accuracy (defined in the next section) with q varying from 0.1 (at least 10% of volunteers should agree to validate a cluster) to 1 (everyone must agree).

Several observations emerge from Fig. 5 (1) As expected due to the correlation between precision and recall, the higher the agreement ratio q , the higher the precision rate (the probability to identify a cluster as a building) and the lower the recall rate (the probability that a building is detected). (2) as q is a factor impacting significantly the global quality of the result (F-measure), it should be chosen carefully. A common method in crowdsourcing is to apply a majority voting rule: a decision is validated if at least more than half of the participants agree i.e. $q = 0.5$. However such rule is clearly not optimal in our case since $q^* \in [0.25, 0.3]$ turns out to give the better F-value.

To explain such a bias in q^* compared to a majority rule, we computes the precision and recall rates at the individual level for all volunteers. We found that the probability for a user to miss a building is in general much higher than to make a mistake when annotating one (i.e. lower recall rate compared to

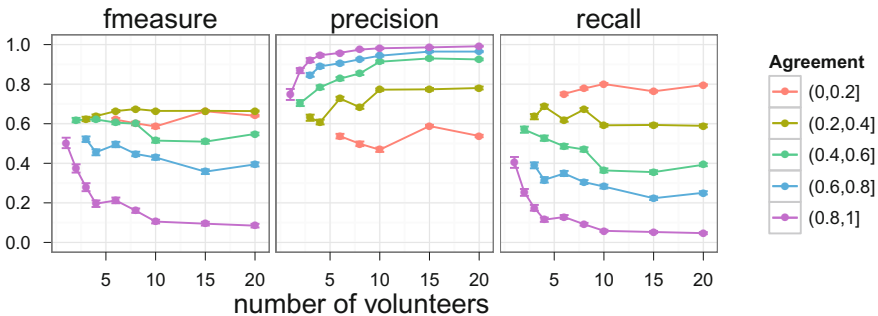


Fig. 5. Accuracy (F-measure, recall, precision) vs. number of volunteers using different agreement ratio q , for map 3

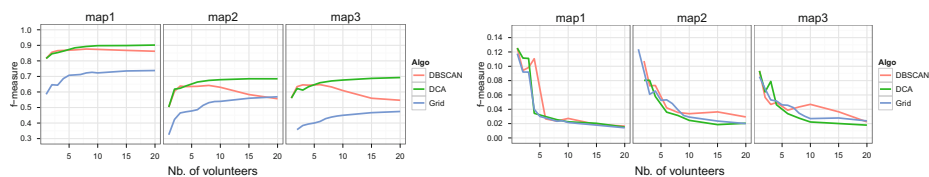
the precision rate). With this bias at the individual level, the optimal decision threshold q^* is lower than 0.5 because the increase of the recall rate is faster than the decrease of precision rate, thus improving the F-measure. In terms of signal theory, this means that the signals generated by the crowd have little noise, so we can lower the filtering level to improve the detection rate.

5.3 Results

In our experiment, n participants have annotated independently the maps. It is $n = 121$ for map 1, $n = 120$ for map 2 and $n = 112$ for map 3. From the n participants, we generate m random groups of k participants. We varied k from 1 to n and chose m as $m = \min(C_n^k, 200)$, where C_n^k is the binomial coefficient. We denote G_k the set of groups with k participants. Then for each group $g \in G_k$ we aggregate the annotations using one of the three clustering algorithms (Grid, DBSCAN, DCA) and compute precision, recall and F-measure of the output, with the best agreement ratio q^* and a tolerance distance of 7 meters. Fig. 6 shows the mean value of the F-measure and the related 95% confidence interval of the distribution for each G_k with $k = 1, \dots, 20$.

First the performance of the DBSCAN is quickly degraded with the amount of volunteers in a dense environment (map 3) compared to sparse environment (map 1). The grid-based algorithm and the DCA are however robust to the chaining effect. The performance of the DCA is the best of the three proposed aggregation methods for all maps.

A second remark is about the existence of a 'skill' threshold. Except for the case of DBSCAN in dense areas, increasing the number of volunteers increases the collective performance (F-measure) in an asymptotic way until a given threshold is reached: around 0.9 for map 1, 0.7 for map 2 and 0.75 for map 3. Allocating more than 5 volunteers shows a limited improvement on the accuracy and variability, while requiring more resources.



(a) Average F-measure of the DBSCAN (red), DCA (green), grid-based clustering (blue). (b) 95% confidence interval of the distribution of the F measure of G_k , $k=[1,20]$ for the DBSCAN (red), DCA (green), grid-based clustering (blue).

Fig. 6. Accuracy and consistency of the parallel model for the three maps and different clustering algorithms

5.4 Analysis of Spatial Inter-agreement

We also investigated the spatial inter-agreement for each pairwise of volunteers (Fig. 7a) to study how the population agree spatially and its correlation with the collective performance. We define the spatial inter-agreement between 2 volunteers V_i and V_j as the set of matching annotations from all the distinct points with a tolerance distance ϵ .

$$Agree_\epsilon(V_i, V_j) = \frac{|V_i \cap_\epsilon V_j|}{|V_i \cup_\epsilon V_j|} \quad (4)$$

In Fig. 7a the inter-agreement between two random users is high for map 1, reflecting the high consensus/low difficulty to analyze the area, as already observed in Fig. 6. Furthermore map 3 and especially map 2 have a lower mean (respectively 0.4 and 0.27) and a higher variance of inter-agreements, in agreement with their increased difficulty, also illustrated in Fig. 6. Thus, the density distribution of the spatial inter-agreement can be used as metrics to better assess and compare the intrinsic difficulty of different areas.

A second post-analysis is to represent the spatial density of type I errors (the location of popular false buildings) and type II errors (the location of commonly missed buildings). For instance a map for type II errors is obtained by coloring each building according to its difficulty i.e. the percentage of volunteers who have identified it. Such a map illustrates qualitatively common pitfalls and differences between non-experts and experts in terms of analysis. Such analysis could be integrated into a teaching method or a collective feedback process to improve the results.

6 Experiment 2: The Iterative Model

The goal of this experiment is to understand the relationship between the number of volunteers or number of iterations, and the accuracy of the output. The instructions for the first iteration are similar to the parallel model. For the subsequent iterations we asked each volunteer to improve the previous iteration by validating or rejecting the annotations, and to create new annotations if required. Every annotation rejected by a volunteer is removed for the next iterations. To avoid spammers the system accepts a contribution only if the first volunteer created more than 50 markers, or if the next user either rejects less than 1/3 of annotations or validates more than 1/3 of them. If the task is not accepted, the system restarts the iteration with another volunteer. A typical sequence of annotation as produced by the sequential model is shown in Fig. 8.

6.1 Results and Comparison with the Parallel Model

The methodology for the iterative model is the following: for each map we generate n independent instances on which we ask volunteers to improve the result. Then we compute, for a given number of iterations (= number of volunteers) the average recall, precision and f-measure rates from all the instances.

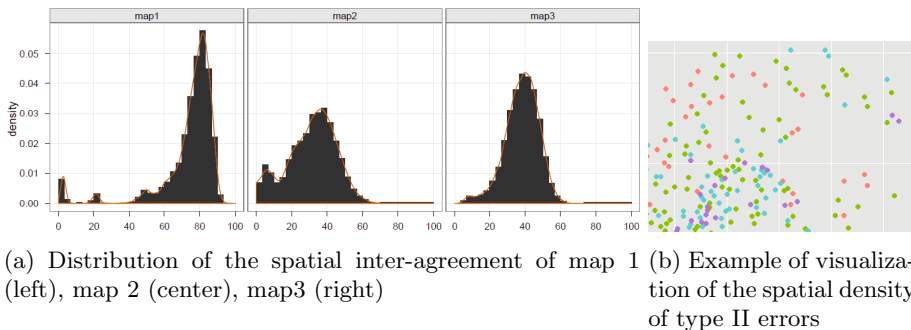


Fig. 7. Post analysis: Fig. 7a represents the distribution of the inter-agreement rate among every pair of volunteers for each map. We clearly see the intrinsic difficulty of analyzing each map though the ratio of disagreement among volunteers. Fig. 7b is an example of spatial representation of the density of type II errors (missed buildings) of an area map 2: each point is colored according to the % of volunteers having identified the building, e.g. purple points represent rarely identified buildings (i.e. by less than 10% of the volunteers), while green ones represent easy identified buildings (>80% of the volunteers).

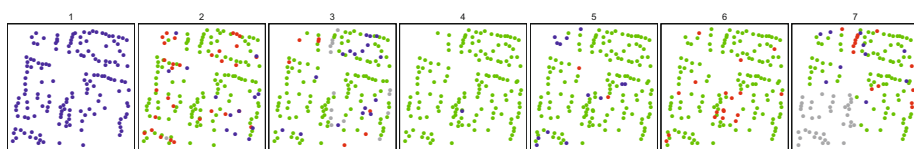


Fig. 8. Example of the iterative process for the map 3: eight iterations are shown, with new markers in blue, validated markers in green, rejected markers in red and not analyzed ones in gray.

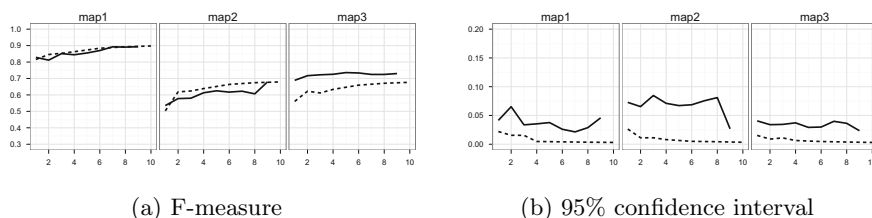


Fig. 9. Basic iterative model (plain line) vs the parallel DCA model (dotted line): average F-measure and 95% confidence interval, as function of the number of volunteers

As data, we generated $n = 13$ instances for map 1 producing from 7 to 10 iterations on each instance (with a total of 107 iterative tasks), $n = 25$ instances for map 2 (with a total of 203 tasks) and $n = 21$ for map 3 (with a total of 174 tasks).

In Fig 9 the F-measure and the related confidence interval are shown, for the iterative strategy (plain line) and the parallel DCA model (dotted line). Several observations emerge: (1) in terms of variability, the redundancy mechanism enables the parallel model to outperform the basic iterative strategy. (2) In terms of accuracy, the iterative strategy outperforms the DCA only for map 3 a dense and difficult area. However, due to our methodology and the limited amount of data produced with only 10 iterations / volunteers, we do not reach the optimal point.

A deeper investigation of the precision and recall rates reveals two main differences between the behavior of the parallel and iterative models. Due to redundancy, the precision rate of any parallel strategy increases with the number of users (less false identification). In an iterative strategy, we observe that the mean precision rate decreases for two out of the three maps: some volunteers did not carefully reviewed the previous work and thus some mistakes accumulated. However, for the recall rate, we observe that an iterative strategy improves the spatial coverage (and thus the recall rate) as the iteration goes on. On the other hand, adding more volunteers in the parallel model does not produce such an effect since additional volunteers work independently and annotate mainly the same obvious buildings.

7 Conclusion

In this paper we investigated the characteristics of two strategies to crowdsource satellite image analysis: the parallel and iterative models. We first analyzed their qualitative differences according to the type of problem, the mechanism enforcing quality, the exploration/exploitation tradeoff and the tasks proposed to the user.

7.1 On the Properties on Each Model

Both models present different forms of redundancy. The parallel model has an horizontal form of redundancy by allocating n independent volunteers in parallel to do the same task, favoring an exploration strategy in the search space. It aggregates the results by keeping consensual decisions (wisdom of the crowd). The iterative model has a vertical form of redundancy by asking n volunteers to perform the same task: improving previous results and thus favoring an exploitation strategy.

parallel model: Linus'law [16], studied for OpenStreetMap, an iterative model, as a limited validity in the parallel model: after a given threshold, adding more volunteers will not change the representativeness of opinion and thus will not change the filtered / consensual output. At this stage, adding more volunteers is not anymore a factor of improvement. In our context, allocating more than 5 volunteers has a low impact on the accuracy and variability, while increasing unnecessary the resources. Furthermore we showed that varying the decision threshold in the voting process is a factor impacting significantly the global quality (F-measure). This threshold should be chosen carefully, especially regarding any bias at the individual level. In our case applying the majority rule produces sub-optimal performance due to such a common bias.

iterative model: We observed that the first iterations have a high impact on the final results due to a path dependency effect: stronger commitment during the first steps are thus a primary concern for using such model (asking expert/committed users to start).

7.2 On the Performance on Each Model

We investigated the quality of both organisational model according to two aspects: the accuracy (type I and type II errors) and consistency of the results. We concluded the following:

Accuracy - type I errors: The parallel strategy, generating only consensual results, corrects type I errors (wrong annotations) more significantly than the iterative model. However in difficult areas (e.g. map 3), it does not mitigate well disagreements. Thanks to the accumulation of knowledge, the iterative model is thus more appropriated to handle ambiguous cases, or problems being hardly divisible in smaller and easier tasks participants will perform better than a parallel model when ambiguous cases are considered to mitigate decision). So the iterative model outperforms the parallel one for difficult/complex areas, but with a potential path dependency effect: mistakes could be propagated, generating more easily type I errors as the iterations proceed.

Accuracy - type II errors: We observed that the iterative model reduces type II errors (the spatial coverage) from one iteration to the next. It outperforms the parallel model due to the accumulation of knowledge, enabling next users to focus their attention on ‘fresh’ areas. The lower spatial coverage that is usually seen with the parallel model is due to the nature of the strategy: due to the independence of the work, the n^{th} volunteer might well annotate for the n^{th} times the same obvious building, without bringing new information at the collective level. This results in a waste of time for the volunteer and the community.

About the consistency of the result: The parallel model provides an output which is more reliable than that of a basic iterative. The reason is that the latter is sensitive to vandalism or knowledge destruction.

According to such findings, an hybrid model could be based on an iterative model to take advantage of knowledge accumulation. But the allocation of volunteers should be driven by the uncertainty of the current annotations. Areas freshly annotated should attract the attention of the next contributors (redundancy) to remove collectively the uncertainty and lock the decision for the remaining iterations. This will avoid knowledge destruction in the next iterations and enable next volunteers to focus only on uncovered areas or other uncertain annotations. The uncertainty of a given ROI can be estimated by the degree of uncertainty of a single volunteer or by the disagreement within a committee of volunteers.

7.3 Tools

In term of methodology and tools, we introduced the use of Mechanical Turk as a fast and cheap simulator in research in VGI to explore and benchmark participatory models. For the parallel model, we proposed a new algorithm to cluster spatial data with democratic constraints. We also investigated the distribution of the spatial inter-agreement and the spatial density of type I and type II errors as (1) a way to assess the intrinsic spatial difficulty of an area and (2) as a training reflexive tool for volunteers to learn from common mistakes (type I and II errors).

Acknowledgment. This research has been conducted within the Citizen Cyberscience Center (<http://www.citizencyberscience.net/>). We acknowledge financial support from a HP Labs Innovation Research Program.

References

1. Baeza-Yates, R., Ribeiro-Neto, B.: *Modern Information Retrieval*, vol. 463. Addison Wesley (1999)
2. Egidi, M., Narduzzo, A.: The emergence of path-dependent behaviors in cooperative contexts. *International Journal of Industrial Organization* 15(6), 677–709 (1997)
3. Ester, M., Xu, X., Kriegel, H.-P., Sander, J.: Density-based algorithm for discovering clusters in large spatial databases with noise, pp. 226–231. *AAAI* (1996)
4. Fang, C., Lee, J., Schilling, M.A.: Balancing exploration and exploitation through structural design: The isolation of subgroups and organization learning. *Organization Science* 21(3), 625–642 (2010)
5. Friess, S.: 50,000 Volunteers Join Distributed Search for Steve Fossett (2007)
6. Hafner, K.: Silicon Valleys High-Tech Hunt for Colleague (2007)
7. Haklay, M., Basiouka, S., Antoniou, V., Ather, A.: How Many Volunteers Does it Take to Map an Area Well? The Validity of Linus’ Law to Volunteered Geographic Information. *The Cartographic Journal* 47(4), 315–322 (2010)
8. Howe, J.: *Crowdsourcing: Why the Power of the Crowd is Driving the Future of Business*, unedited edition. Crown Business (2008)
9. Kanefsky, B., Barlow, N.G., Gulick, V.C.: Can distributed volunteers accomplish massive data analysis tasks? *Lunar and Planetary Science* 32, 1272 (2001)
10. Lazer, D., Friedman, A.: The network structure of exploration and exploitation. *Administrative Science Quarterly* 52(4), 667–694 (2007)
11. Lorenz, J., Rauhut, H., Schweitzer, F., Helbing, D.: How social influence can undermine the wisdom of crowd effect. *Proceedings of the National Academy of Sciences of the United States of America* 108(22), 9020–9025 (2011)
12. Malone, T.W., Laubacher, R., Dellarocas, C.: Harnessing crowds: Mapping the genome of collective intelligence. MIT Center for Collective Intelligence (No. 4732-09), 1–20 (2009) (retrieved June 10, 2009)
13. March, J.G.: Exploration and exploitation in organizational learning. *Organization Science* 2(1), 71–87 (1991)
14. Mason, W.A.: *How to use mechanical turk for cognitive science research*, New York (2011)

15. Quinn, A.J., Bederson, B.B.: A taxonomy of distributed human computation. HumanComputer Interaction Lab Tech Report University of Maryland (2009)
16. Raymond, E.: The cathedral and the bazaar. *Knowledge Technology Policy* 12(3), 23–49 (1999)
17. ImageCat RIT, World Bank, GFDRR. Remote Sensing and Damage Assessment Mission Haiti (2010)
18. Snow, R., O'Connor, B., Jurafsky, D., Ng, A.Y.: Cheap and fast but is it good? evaluating non-expert annotations for natural language tasks. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 254–263 (October 2008)
19. Surowiecki, J.: *The wisdom of crowds: why the many are smarter than the few and how...* Doubleday (2004)
20. Welinder, P., Branson, S., Belongie, S., Perona, P.: The Multidimensional Wisdom of Crowds. *Most* 6(7), 1–9 (2010)
21. Whitehill, J., Ruvolo, P., Wu, T., Bergsma, J., Movellan, J.: Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. *Security* (1), 1–9
22. Woolley, A.W., Chabris, C.F., Pentland, A., Hashmi, N., Malone, T.W.: Evidence for a collective intelligence factor in the performance of human groups. *Science* 330(6004), 686–688 (2010)

Quantifying Resolution Sensitivity of Spatial Autocorrelation: A Resolution Correlogram Approach

Pradeep Mohan*, Xun Zhou*, and Shashi Shekhar

University of Minnesota, Minneapolis, USA
{mohan,xun,shekhar}@cs.umn.edu

Abstract. Raster spatial datasets are often analyzed at multiple spatial resolutions to understand natural phenomena such as global climate and land cover patterns. Given such datasets, a collection of user defined resolutions and a neighborhood definition, resolution sensitivity analysis (RSA) quantifies the sensitivity of spatial autocorrelation across different resolutions. RSA is important due to applications such as land cover assessment where it may help to identify appropriate aggregations levels to detect patch sizes of different land cover types. However, Quantifying resolution sensitivity of spatial autocorrelation is challenging for two important reasons: (a) absence of a multi-resolution definition for spatial autocorrelation and (b) possible non-monotone sensitivity of spatial autocorrelation across resolutions. Existing work in spatial analysis (e.g. distance based correlograms) focuses on purely graphical methods and analyzes the distance-sensitivity of spatial autocorrelation. In contrast, this paper explores quantitative methods in addition to graphical methods for RSA. Specifically, we formalize the notion of resolution correlograms(RCs) and present new tools for RSA, namely, rapid change resolution (RCR) detection and stable resolution interval (SRI) detection. We propose a new RSA algorithm that computes RCs, discovers interesting RCRs and SRIs. A case study using a vegetation cover dataset from Africa demonstrates the real world applicability of the proposed algorithm.

Keywords: Resolution correlograms, descriptive correlogram statistics, rapid change resolution, stable resolution intervals, resolution sensitivity analysis.

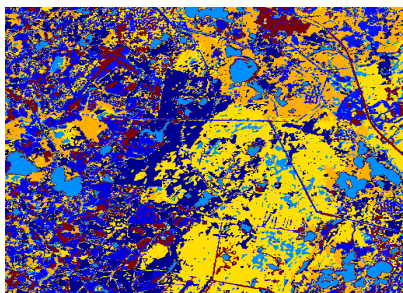
1 Introduction

Many raster spatial datasets exhibit spatial autocorrelation [8,5,11,9,6,16,15], a unique property recognized by Tobler’s famous observation that “all things are related, but nearby things are more related” [16]. GIScience research tradition has christened this observation the *First law of geography* and echoed the need

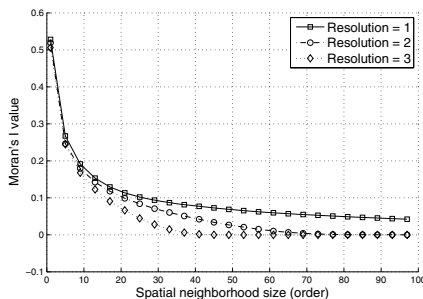
* Corresponding authors.

to honor spatial autocorrelation while formulating analytical methods for spatial data analysis [9,6]. A common application of spatial autocorrelation is land cover assessment [12,10,3,4,19], where it is useful in analyzing raster datasets to determine the natural patch size of the underlying land cover (e.g., patches of plant growth in mountainous regions [12]). Here, spatial autocorrelation analysis provides insights into similarities and differences between different land cover types within and across eco-regions [12,3,4]. This analysis helps scientists incorporate spatial dependence structures into multiple regression models that may help predict land cover dynamics [3,4].

However, spatial autocorrelation has been observed to be sensitive to spatial parameters such as neighborhood definition, e.g., used to define spatial weights matrix [12,3,4,12]. To illustrate, we consider a simple example from land cover assessment. Figure 1(a) shows a sample land cover image dataset where individual cell values is the mean of three bands (e.g. Red, Green and Blue). Figure 1(b) shows the sensitivity of spatial autocorrelation value as measured by the Moran's I spatial autocorrelation measure. There are three different spatial neighborhood sensitivity plots in this figure, each corresponding to a different spatial resolution of the input image (Figure 1(a)). Each of the sensitivity plots in Figure 1(b) is a *spatial correlogram* [5]. Since these plots reveal the variation in spatial autocorrelation across several neighborhood sizes, these plots can be termed as *distance based correlograms*.



(a) Input Image at 512×512 Resolution (Mean of the three bands)



(b) Distance sensitivity of spatial autocorrelation (fixed resolution)

Fig. 1. Sample Land cover dataset (Best Viewed in Color)

Even though Figure 1(b) reveals the distance based sensitivity of spatial autocorrelation, many applications, particularly land cover assessment require tools that can quantify the sensitivity of spatial autocorrelation (e.g. by detecting useful patterns of variation in spatial autocorrelation) with respect to the resolution (cell size) of the spatial dataset. Such an analysis is particularly useful to produce datasets where the presence or absence of certain patterns in natural phenomena may be dictated by the cell size in the dataset (as observed in existing literature [14,11,2,3,4]). In this application context, quantifying resolution sensitivity becomes important to provide additional insights into the patterns of variation in

spatial autocorrelation [19]. Some important questions posed by this application include, *Are there resolutions where spatial autocorrelation changes rapidly? Are there resolution ranges where spatial autocorrelation is stable?* Performing resolution sensitivity analysis (RSA) on raster datasets generated from applications like land cover assessment and climate science may help answer such questions, allowing users to make informed decisions about the extent of cell aggregation.

In typical application scenarios, users provide a raster spatial dataset similar to the one shown in Figure 1(a) at different spatial resolutions or cell sizes. In addition, information pertaining to a fixed spatial neighborhood and a threshold on the sensitivity of spatial autocorrelation are also provided. Given these inputs, the goal of RSA is to quantify the variability in spatial autocorrelation by reporting interesting patterns in this variability. For example, given a raster dataset such as Figure 1, RSA can identify rapid change resolutions (RCRs) and discover stable resolution intervals (SRIs) by computing *resolution correlograms* (RCs) as opposed to distance correlograms. By identifying these interesting patterns, RSA can provide additional insights for choosing appropriate aggregation levels for raster datasets.

However, performing RSA is challenging for two key reasons: (a) the absence of a multi-resolution definition of spatial autocorrelation and, (b) possible non-monotone variation in spatial autocorrelation across resolutions. For example, the value of Moran's I can change in either direction (positive or negative), implying that overall decreasing trends in autocorrelation may also include increasing trend subsets.

Related Work: Approaches to scale sensitivity analysis of spatial autocorrelation fall broadly into two categories: (a) distance sensitivity analysis (DSA) [12,3,15] and (b) Resolution sensitivity analysis (RSA). Figure 2 shows the classification of different approaches to scale sensitivity analysis of spatial autocorrelation. The left category, DSA is based on well known methods such as distance based correlograms that are primarily graphical methods. A graphical method of sensitivity analysis may be helpful to understand the overall trend in the variability of spatial autocorrelation. However, such methods are limited in quantitative reasoning and do not reveal interesting insights regarding different patterns of change in spatial autocorrelation. Our work addresses this gap by focusing on a quantitative methodology that provides new insights by discovering patterns of change in spatial autocorrelation in addition to providing a basic graphical representation of resolution sensitivity.

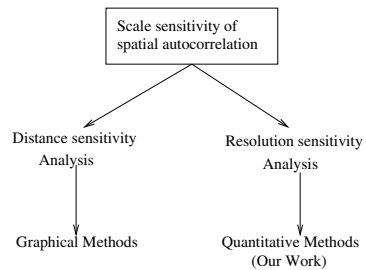


Fig. 2. Classification of Related Work

Our Contributions: Specifically, this paper makes the following contributions: (a) We define the Resolution sensitivity analysis problem; (b) We formalize the notion of resolution correlograms (RCs) for two popular spatial autocorrelation measures, namely, Moran’s I [11] and Geary’s C [8]; (c) We provide simple examples via descriptive resolution correlogram statistics that can describe simple trends in spatial autocorrelation across resolutions; (d) We propose a novel RSA algorithm that can compute resolution correlograms, descriptive correlogram statistics and discover interesting patterns, including rapid change resolutions(RCRs) and stable resolution intervals(SRIs);(d)Finally, we provide a case study using a GIMMS vegetation cover dataset from Africa to validate the usefulness of RSA in a real world application setting.

Scope: While the notion of resolution sensitivity can be applied to vector datasets as well, this paper focuses primarily on raster data. The notion of scale can have multiple meanings and definitions. This paper primarily focuses on spatial resolution as a form of scale. Detailed performance evaluation of the RSA algorithm is beyond the scope of this paper. Also, the aim of the case study here is to demonstrate the real world applicability of proposed approaches. A detailed domain interpretation of discovered patterns or insights is beyond the scope of the current work. Finally, the RSA problem described in this paper relies on the user to input data pertaining to different resolutions or to specify a meaningful aggregation scheme for pixels. For simplicity, we make use of a pixel aggregation based mean of neighboring pixels. Other aggregation schemes are beyond the scope of this paper.

Convention: Resolution in raster datasets is usually referred to as the inverse of cell size. In this paper, references to change in resolution implies any change in cell size. Spatial neighborhood can be specified via topological notions such as queen connectivity or via a fixed spatial lag. However, examining differences between use of either spatial neighborhood definition is beyond the scope of this paper.

Outline: The paper is organized as follows: (a) Section 2 formalizes the notion of resolution correlograms and formulates the RSA problem; (b) Section 3 outlines the general layout of the RSA algorithm and describes specific details of RCR detection and SRI discovery; (c) Section 4 evaluates the real world applicability of RSA on a vegetation cover dataset from Africa and reports potentially interesting trends in spatial autocorrelation for this dataset; (d) Section 5 discusses several issues relevant to RSA, including its relationship with other forms of multi-resolution analysis; (f) Finally, Section 6 concludes the paper.

2 Basic Concepts and Problem Statement

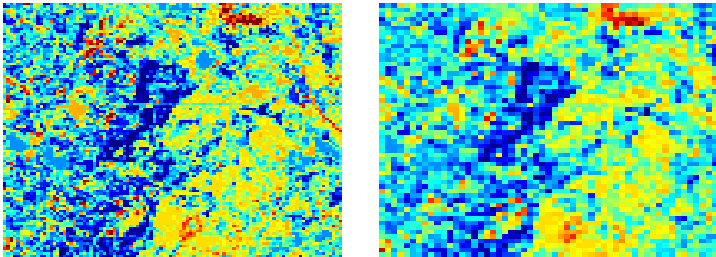
This section reviews several basic concepts, formalizes the notion of resolution correlograms, presents descriptive correlogram statistics and formulates the RSA problem.

2.1 Basic Concepts

A **raster spatial dataset** is a sample of continuous natural phenomena as observed by a data collection system such as a sensor or a satellite. Raster datasets consist of simple units called **pixels or picture cells**. Many raster datasets are characterized as matrices of cells and are labeled by the number of cells contained in the rows and columns of this matrix. Figure 1(a) shows a raster dataset with rows and columns containing 512 cells each.

Each cell within a raster dataset may contain some information about a phenomenon that was observed by a data capturing system. This information may sometimes be distributed across several layers or bands. For example, the raster dataset in Figure 1(b) contains aggregate information, that is, the mean of three individual color bands namely, red, green and blue.

The **Resolution** of a raster dataset is determined based on the physical size of cells, e.g., $1m \times 1m$ or $1km \times 1km$ or $1^\circ \times 1^\circ$ latitude, longitude. Due to the availability of multiple data layers, scientists may choose to create raster datasets at different spatial resolutions of a spatial dataset to understand a natural phenomenon such as land cover. For example, Figure 3 shows the dataset in Figure 1(a) at two different resolutions. The aggregation process usually involves combining neighboring cells to form cells of larger size. For example, in Figure 1(a) the aggregation of 5 neighboring cells yields an aggregated dataset (Figure 3(a)) where cells sizes are 5 times those in the original dataset. Similarly, combining 9 neighboring cells yields a coarser dataset as shown in Figure 3(b). It is apparent from Figure 3 that as the cell size increases, the cell level information changes. An important consequence of this is the possible change in cell neighborhood information.



(a) Aggregated image (cell size = 5) (b) Aggregated image (cell size = 9)

Fig. 3. A raster dataset at multiple resolutions(Best Viewed in Color)

In spatial analysis, neighborhood information is usually represented as a **spatial weights matrix** denoted as **W** [1]. Cell aggregation in raster datasets essentially changes the original dataset and disturbs the W-Matrix, making it sensitive to a change in spatial resolution. This change in the structure of the W-Matrix makes all spatial autocorrelation analysis techniques sensitive to changes in spatial resolution. Traditional spatial autocorrelation measures such as Moran's I

and Geary's C are particularly sensitive to changes in data resolution as well as change in the W-Matrix, especially the row normalized W-Matrix.

When the resolution of a raster dataset is varied via aggregation, the sensitivity in Moran's I and Geary's C with respect to resolution can be graphically represented as a **Resolution Correlogram** as illustrated in Figure 4. Figure 4(a) and (b) show the Moran resolution correlogram (MRC) and the Geary resolution correlogram (GRC) respectively derived from the image dataset of Figure 1(b). The X-axis in Figure 4(a) and (b) shows different cell sizes corresponding to different resolution levels. The Y-axis shows the spatial autocorrelation level at the corresponding resolution measured using either Moran's I or Geary's C. In addition to a resolution correlogram, many users may be interested in descriptive statistics that can provide insights on the distribution of spatial autocorrelation across resolution.

Descriptive resolution correlogram statistics provide a quantitative summary of spatial autocorrelation sensitivity across resolutions.

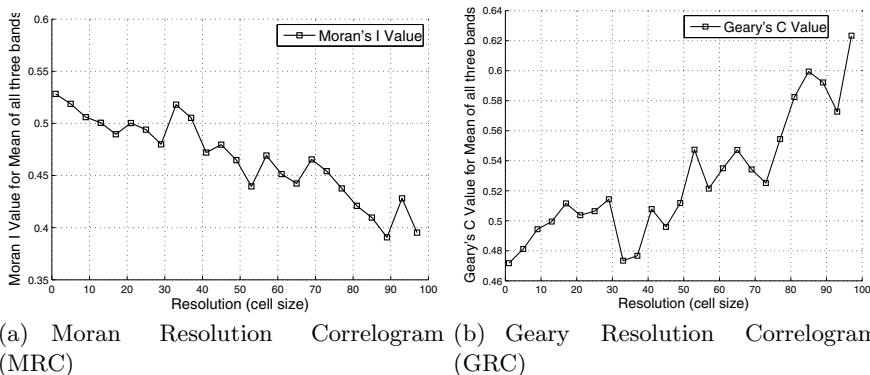


Fig. 4. Resolution correlograms for Image dataset in Figure 1(b)

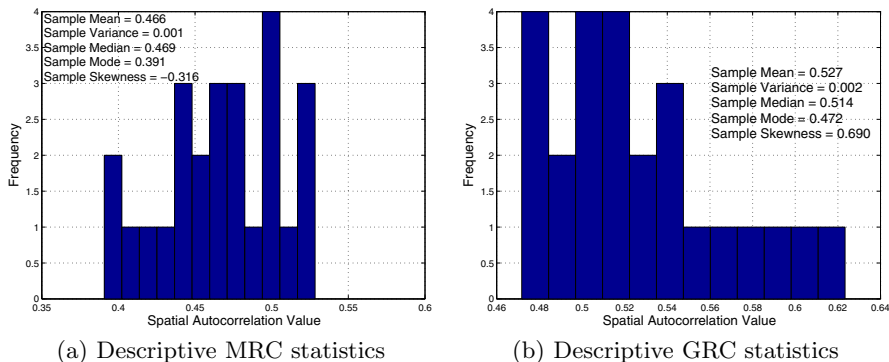


Fig. 5. Descriptive resolution correlogram statistics

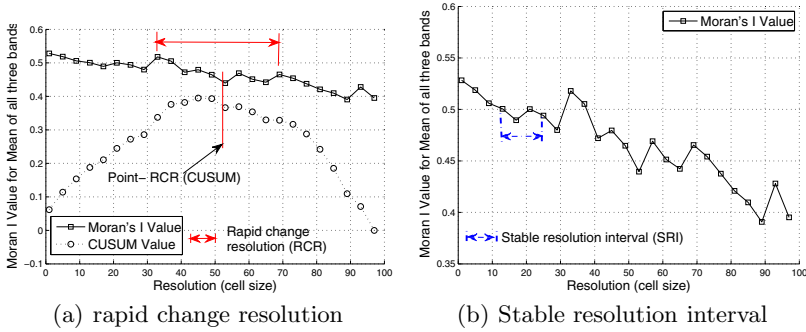


Fig. 6. Resolution change patterns in spatial autocorrelation

For example, Figure 5(a) and (b) show the histogram with descriptive resolution statistics for the MRC and the GRC respectively. In Figure 5(a) the sample shows a negative skew with the median autocorrelation at 0.469. Also, the sample variance is significantly low (e.g., 0.001) indicating a possible positive spatial autocorrelation across all resolutions. A similar trend is observed in Figure 5(b) for the descriptive GRC statistics. However, descriptive statistics may provide only a summary view of resolution sensitivity. For deeper insight, one may want to detect interesting, useful and non-trivial patterns from resolution correlograms. Two such patterns are: (a) rapid change resolution and (b) stable resolution interval .

A **rapid change resolution (RCR)** represents a pattern of rapid increase or decrease in spatial autocorrelation across resolutions. RCR can be a **change point** or **change interval**. For example, Figure 6(a) shows RCRs computed from the MRC highlighted as red ellipses. RCRs can be quantified using several schemes, including the CUSUM statistic [13] and other measures based on the rate of change in spatial autocorrelation across resolutions [20]. In this figure, the curve with a thick black line and points represented as squares is the MRC. The second curve with dotted lines and data points represented by circles is the CUSUM statistic for the MRC. Figure 6(a) shows dotted red ellipses highlighting RCRs corresponding to intervals and points respectively. The CUSUM value corresponding to the point RCR is also highlighted. While RCRs can help identify resolutions of unstable spatial autocorrelation, discovering resolutions of stable autocorrelation may help scientists make informed decisions to appropriately choose the correct spatial resolution for analysis. A **stable resolution interval (SRI)** can be defined as a collection of resolutions at which spatial autocorrelation is relatively unchanging. For example, Figure 6(b) shows SRIs for the MRC highlighted by dotted blue ellipses.

Problem Statement : Based on the above concepts, we define the RSA problem as follows:

Input: (a) a raster spatial dataset at multiple resolutions; (b) a sensitivity threshold and ; (c) a fixed spatial neighborhood definition.

Output: (a) resolution correlograms (e.g. MRC and GRC); (b) descriptive resolution correlogram statistics; (c) rapid change resolutions; (d) stable resolution intervals.

Constraints: Correctness and Completeness.

Example: In land cover assessment, the input raster spatial dataset may be available at different resolutions as shown in Figure 3. The sensitivity threshold can be defined as a standard deviation threshold (e.g. 0.005). The fixed spatial neighborhood can be defined as a topological neighbor (e.g. queen connectivity) or lag based neighbor (e.g. 1000 meters). Based on these inputs, the goal of the RSA problem is to find the following outputs: (a) Resolution correlograms, MRC and GRC as shown in Figure 4; (b) descriptive resolution correlogram statistics as shown in Figure 5; (c) rapid change resolution as shown in Figure 6(a); and (d) stable resolution intervals, as shown in Figure 6(b).

3 Resolution Sensitivity Analysis Algorithm

In this section we describe the general structure of our resolution sensitivity analysis (RSA) algorithm, including steps for detecting intervals of stable resolution and rapid change. The RSA algorithm has four important steps: (a) Resolution correlogram computation, (b) Descriptive correlogram statistics computation, (c) rapid change resolution detection and, (d) stable resolution interval discovery.

Detailed explanation of steps in RSA algorithm:

Algorithm 1. RSA Algorithm

Input: (a) *Raster Spatial Dataset at different resolutions.*

(b) *Spatial neighborhood size.*

(c) *A sensitivity threshold.*

Output: (a) *Resolution Correlogram for Moran's I and Geary's C.*

(b) *Descriptive correlogram statistics.*

(c) *Abrupt change resolutions.*

(d) *Stable resolution intervals.*

- 1: *Initialize MRC, GRC, MRW*
 - 2: **for** $r := 1 \rightarrow \text{maxResolution}$ **do**
 - 3: *Compute W – Matrix, $W(r)$*
 - 4: $MRW \leftarrow MRW \cup W(r)$
 - 5: *Compute Moran's I, $I(r)$ using $W(r)$*
 - 6: $MRC \leftarrow MRC \cup I(r)$
 - 7: *Compute Geary's C, $C(r)$ using $W(r)$*
 - 8: $GRC \leftarrow GRC \cup C(r)$
 - 9: **end for**
 - 10: *Compute Descriptive Correlogram Statistics*
 - 11: *Compute Rapid Changes (RCR)*
 - 12: *Compute Stable Resolution Intervals(SRI)*
-

Steps 2-9 of the algorithm compute the Moran Resolution Correlogram (MRC) and the Geyary Resolution Correlogram (GRC). To do this, the algorithm computes the W-Matrix, $W(r)$ corresponding to a resolution r . For example, given a dataset such as the one in Figure 3 and suitable thresholds, these steps compute MRC and GRC as shown in Figure 4.

Step 10 computes different descriptive correlogram statistics, including, the sample correlogram mean, sample variance, sample median, sample skewness, and the sample mode. This step also computes a histogram to represent the population of spatial autocorrelation values.

Step 11 discovers interesting patterns of change in spatial autocorrelation, namely, rapid change. In this step, these are points and intervals where a spatial autocorrelation value that undergoes a sharp increase or decrease is reported.

Step 12 discovers other interesting trends in spatial autocorrelation, namely, resolution intervals where spatial autocorrelation is stable within a sensitivity threshold.

Steps 11 and 12 of the RSA algorithm report interesting, useful and non-trivial trends in spatial autocorrelation based on different resolution correlograms. We provide additional details of these steps in the next section.

3.1 Discovering Spatial Autocorrelation Trends

The RSA algorithm reports two types of patterns in spatial autocorrelation: (a) rapid change resolution (RCR) and (b) stable resolution intervals (SRI) based on the computed resolution correlograms and user specified sensitivity thresholds. In step 11, the algorithm computes RCRs that include both points and intervals. To detect point RCRs, RSA computes the CUmulative SUM (CUSUM) [13] statistic and reports any resolution that may show a rapid change in spatial autocorrelation value with respect to the mean level. To compute interval RCRs, the RSA algorithm evaluates the persistence of any rapid change within a resolution interval. Since it is also likely that the autocorrelation is non-monotone within a resolution interval, the RSA algorithm makes use of a statistic that is based on the average change in autocorrelation across resolutions to quantify the rate of change. The algorithm picks the RCRs that exceed the top α percentile in slope as “rapid change units.” The score of the statistics for each candidate interval RCR is computed as follows: $score = \frac{AVG(all \ \Delta I)}{AVG(\Delta I \ of \ rapid \ change \ units)}$

It can be proved that the value is between 0 and 1, where a larger value indicates that the overall change trend is more likely to be rapid. For a given sensitivity threshold, the algorithm finds all the resolution intervals that have a score that is larger than this threshold, and eliminates shorter intervals that are subsets of any RCR. In the running example of Figure 6(b), we discovered a number of such resolution intervals, including the one from pixel size 33 to 57. A detailed explanation of different enumeration schemes to find the RCR efficiently was explored in our earlier paper [20].

Figure 7 illustrates the RCR computation step. The boxes represent all the resolutions and corresponding intervals. The diagonal line represents individual

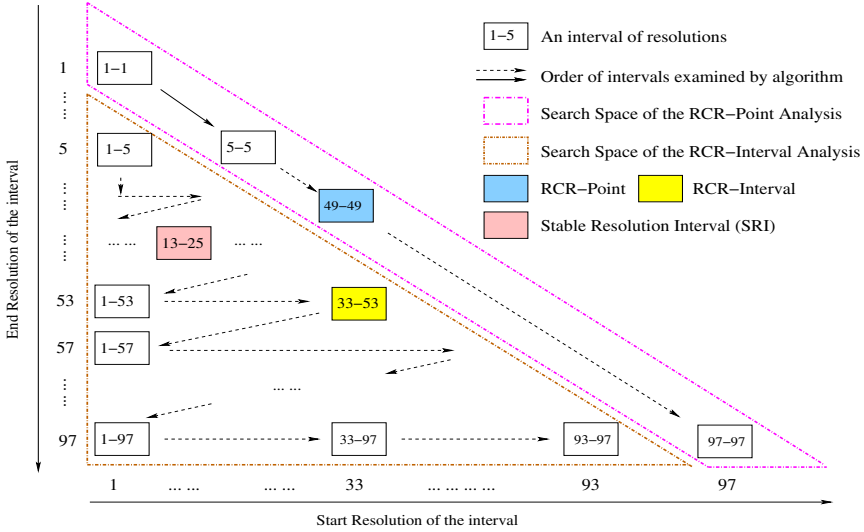


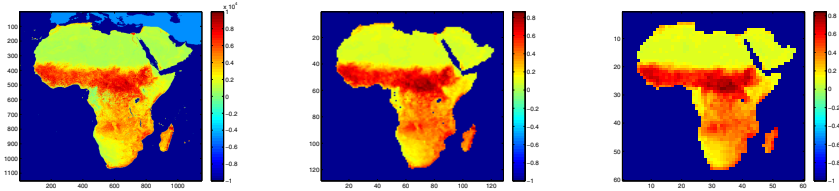
Fig. 7. Discovering interesting spatial autocorrelation trends(Best viewed in Color)

resolutions. To find a resolution change point, RCR analysis examines all diagonal candidates sequentially and keeps a score until the change point is discovered. To find resolution intervals, the RCR discovery step traverses the entire space of resolution intervals, and computes the score for each one. The figure illustrates a row-wise enumeration strategy where all the intervals ending with the same resolution are examined from longer ones to shorter ones. Detailed pseudo code can be found in our related paper [20].

In step 12, the RSA algorithm computes other interesting resolution ranges (e.g. stable resolution intervals) by making use of an altered score function that can be written as follows: $score = SQRT(AVG(X_i^2) - (AVG(X))^2)$, where $AVG(X_i^2)$ and $(AVG(X))^2$ can be computed using simple functions such as $SUM()$ and $COUNT()$. Techniques such as building lookup tables can be used to further accelerate the computations. Details can be found in our related paper [20]. As shown in Figure 6, we found that the Moran’s I value is stable at resolutions ranging from 13 to 25. Also, Figure 7 shows that stable resolution intervals (SRIs) can also be computed in a manner similar to RCRs. The SRIs enumerated are shown as red colored boxes in Figure 7.

4 Case Study

We illustrate the usefulness of RSA through a real world case study on a GIMMS vegetation cover dataset in Africa [17]. Figure 8(a) shows one snapshot of this dataset in August, 1981. Data values are the Normalized Difference Vegetation Index (NDVI) measured between 0 and 1. A larger value indicates more vegetation cover on the ground. Ocean and areas outside the study region are marked with invalid value, and were ignored in the analysis. The dimension of



(a) Input data (original resolution) (b) Aggregated data (resolution at 0.35 degrees) (c) Aggregated data (resolution at 0.7 degrees)

Fig. 8. Input data at different resolutions(Best in Color)

this dataset is 1152 by 1152 pixels, where each original pixel represents about 0.07 degree on the earth surface. Figure 8 shows one snapshot of this dataset at three different resolutions. We applied the RSA algorithm on four snapshots in the dataset, namely, August 1981, November 1981, February 1982 and May 1982, and present the results. These four snapshots of vegetation cover correspond to four different seasons in Africa. This analysis was performed using Matlab 2010b on a 4 Core 2.53G Workstation with a Ubuntu Linux system.

4.1 Method and Results

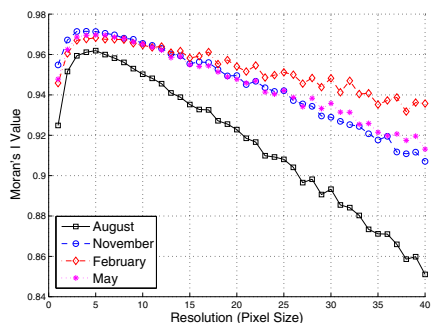
For this study, we ran the RSA algorithm to compute both the MRC and GRC for 40 different resolutions of the GIMMS vegetation dataset, with cell sizes ranging from 1 to 40 times the original size. Aggregated images for resolutions 10 and 20 are shown in Figure 8(b) and Figure 8(c). After computing the resolution correlograms, the algorithm also identified RCR intervals and SRIs from these correlograms. Figure 9 shows the results of applying the RSA on the four different seasons of GIMMS vegetation dataset at 40 different resolutions. Figure 9(a) and Figure 9(c) show the MRC and the GRC respectively for the four seasons. The general trend observed in these figures is that the spatial autocorrelation measured by Moran's I increases at very fine resolutions, reaching a peak and then slowly drops to a lower level. This suggests that the data contains a certain level of local heterogeneity. The turning point of the curve shows the resolution at which autocorrelation or heterogeneity vanish.

This analysis helps in data preprocessing and noise smoothing. However, for different seasons, the turning points vary. As shown in Figure 9(a), in summer (August), the data is more locally heterogeneous as the Moran's I value reaches its maximum at a coarser resolution. Analogous trends are also observed for the Geary's C measure in Figure 9(c) where the value of Geary's C decreases and then increases. This can be interpreted as an increase and then subsequent decrease in spatial autocorrelation with respect to resolution.

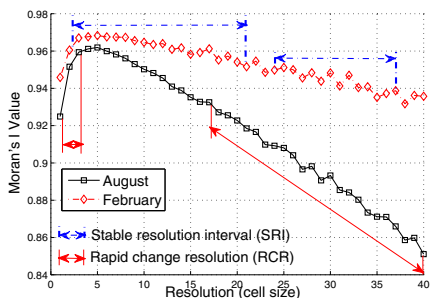
Figure 9(b) and 9(d) show different patterns in spatial autocorrelation sensitivity detected by RSA. The thick red lines in Figure 9(b) and 9(d) show the RCR intervals and the dotted blue lines show the SRIs. Among the four seasons, the ones for February (winter) and August (summer) show interesting results.

With the sensitivity threshold set to $|\Delta I| \geq 0.005$ and a score threshold of 0.5, we find that the curve for February has hardly any RCR intervals while the August curve has a steadily decreasing interval from resolution 17 to 40 (as shown in Figure 9(b)). With the same sensitivity threshold value of 0.005, the Moran's I value for the February curve stays stable from resolution 2 to 21 and from 23 to 37, while there is no stable interval found in the August curve using the same threshold. As for the Geary's C measure, even though the trends are opposite, the interpretation is analogous.

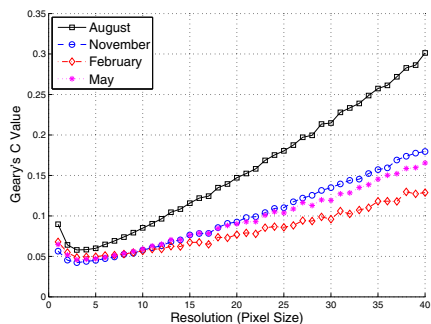
These results are likely due to the fact that the vegetation in the rain forest and grassland are less irrigated by precipitation in the dry winter, which brings down the spatial heterogeneity at large scales. By contrast, the dense rain forest and grassland in summer (August) makes the land cover quite different at large scales, compared to the large area of deserts in the north.



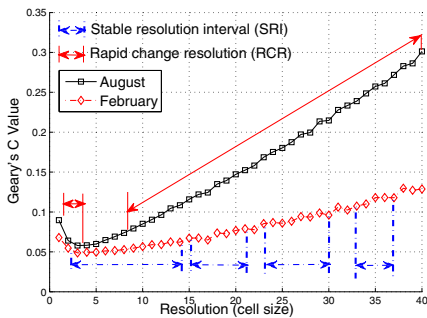
(a) The Moran Correlogram of Africa in August, November 1981 and February, May 1982.(Best Viewed in Color)



(b) Stable and rapid change resolution intervals of the August and February Moran Correlograms(Best Viewed in Color)



(c) The Geary Correlogram of Africa in August, November 1981 and February, May 1982.(Best Viewed in Color)



(d) Stable and rapid change size resolution intervals of the August and February Geary Correlograms(Best Viewed in Color)

Fig. 9. Results of RSA on GIMMS vegetation cover dataset from Africa

5 Discussion

In this section we briefly discuss several issues relevant to RSA including, first steps to new research directions and multi-resolution tools that do not directly analyze spatial autocorrelation sensitivity.

The RSA algorithm proposed in this paper does not explore any computationally efficient schemes to guarantee better computational performance. Exploring new computational approaches that can speed up the performance of RSA may be another interesting direction for research. For example, in the current formulation of the algorithm, the W-Matrix is computed repeatedly. However, similar to data transformation via pixel aggregation, it may also be possible to define new W-transformation techniques that are cheaper than re-computing W itself. This might be a promising direction, particularly in cases where the spatial neighborhood sizes are large. Also, from a spatial database perspective, W computation can be viewed as a spatial join [15]. The spatial database literature has explored a vast family of spatial indices (e.g. Quad Tree, R-Tree) that may be useful for W computation.

While this paper focuses on multi-resolution sensitivity of spatial autocorrelation, multi-resolution analysis itself is a well studied area, particularly dominated by a family of mathematical structures called Wavelets [18,7]. Wavelets are primarily parametric methods which make use of a fixed set of basis functions to model observations of a natural phenomena that may exist across multiple resolutions. In addition, wavelet based methods assume a fixed aggregation hierarchy for pixels in powers of two, which may or may not represent the reality. In contrast, this paper explored non-parametric methods (e.g., SRI discovery) to discover interesting patterns in resolution sensitivity of spatial autocorrelation.

6 Conclusion

This paper explored the resolution sensitivity of spatial autocorrelation in the context of land cover assessment. This paper formalized the notion of resolution correlograms based on the popular Moran's I and Geary's C measures of spatial autocorrelation. We introduced a new resolution sensitivity analysis algorithm that computes these correlograms, descriptive correlogram statistics and reports interesting patterns of change in spatial autocorrelation. Finally, a case study using the GIMMS vegetation cover dataset from Africa validated the real world applicability of resolution sensitivity analysis. In resolution sensitivity analysis, sometimes it may be useful to aggregate pixels via clusters. In future work, we hope to explore the effect of different aggregation schemes including clustering to generate datasets of coarser resolution. The approach for resolution sensitivity analysis proposed here utilizes global autocorrelation statistics. However, local autocorrelation statistics might provide an enhanced view of variability in spatial autocorrelation levels across resolutions. Hence, in future work, we plan to explore analysis of local resolution sensitivity.

Acknowledgement. We thank the members of the spatial database and data mining group for their feedback on the initial versions of this paper. We also thank Kim Koffolt for helping to improve the readability of this paper. This work was supported by grants from the US Army (W9132V-09-C-0009) and NSF Expeditions in Computing (No. 1029711).

References

1. Anselin, L.: Under the hood issues in the specification and interpretation of spatial regression models. *Agricultural Economics* 27(3), 247–267 (2002)
2. Atkinson, P.M., Tate, N.J.: Spatial scale problems and geostatistical solutions: A review. *The Professional Geographer* 52(4), 607–623 (2000)
3. de Koning, G., Veldkamp, A., Fresco, L.: Land use in ecuador: a statistical analysis at different aggregation levels. *Agriculture, Ecosystems and Environment* 70(2-3), 231–247 (1998)
4. de Koning, G., Verburg, P., Veldkamp, A., Fresco, L.: Multi-scale modelling of land use change dynamics in ecuador. *Agricultural Systems* 61(2), 77–93 (1999)
5. Ebdon, D.: *Statistics in geography*. Blackwell Publisher (1985)
6. Fischer, M., Getis, A.: *Handbook of applied spatial analysis: software tools, methods and applications*. Springer (2010)
7. Foufoula-Georgiou, E., Kumar, P.: *Wavelets in geophysics. Wavelet analysis and its applications*, vol. 4. Academic Press (1994)
8. Geary, R.C.: The contiguity ratio and statistical mapping. *The Incorporated Statistician* 5(3), 115–127, 129–146 (1954)
9. Goodchild, M.F.: The validity and usefulness of laws in geographic information science and geography. *Annals of the Association of American Geographers* 94(2), 300–303 (2004)
10. Ju, J., Gopal, S., Kolaczyk, E.: On the choice of spatial and categorical scale in remote sensing land cover classification. *Remote Sensing of Environment* 96(1), 62–77 (2005)
11. Moran, P.A.P.: The interpretation of statistical maps. *Journal of the Royal Statistical Society. Series B (Methodological)* 10(2), 243–251 (1948)
12. Overmars, K., de Koning, G., Veldkamp, A.: Spatial autocorrelation in multi-scale land use models. *Ecological Modelling* 164(2-3), 257–270 (2003)
13. Page, E.: Continuous inspection schemes. *Biometrika* 41(1-2), 100–115 (1954)
14. Quattrochi, D., Goodchild, M.: *Scale in remote sensing and GIS. Mapping Sciences Series*. Lewis Publishers (1997)
15. Shekhar, S., Xiong, H.: *Encyclopedia of GIS*. Springer Reference. Springer (2008)
16. Tobler, W.R.: A computer movie simulating urban growth in the detroit region. *Economic Geography* 46, 234–240 (1970)
17. Tucker, C., Pinzon, J., Brown, M.: Global inventory modeling and mapping studies (gimms) satellite drift corrected and noaa-16 incorporated normalized difference vegetation index (ndvi), monthly 1981–2002. Global Land Cover Facility, University of Maryland (2004)
18. Willsky, A.: Multiresolution markov models for signal and image processing. *Proceedings of the IEEE* 90(8), 1396–1458 (2002)
19. Woodcock, C.E., Strahler, A.H.: The factor of scale in remote sensing. *Remote Sens. Environ.* 21(3), 311–332 (1987)
20. Zhou, X., Shekhar, S., Mohan, P., Liess, S., Snyder, P.K.: Discovering interesting sub-paths in spatiotemporal datasets: a summary of results. In: *GIS*, pp. 44–53 (2011)

LocalAlert: Simulating Decentralized Ad-Hoc Collaboration in Emergency Situations

Silvia Nittel, Christopher Dorr, and John C. Whittier

Spatial Informatics, School of Computing and Information Science
University of Maine, Orono, USA
nittel@spatial.maine.edu,
{christopher.h.dorr,john.c.whittier}@maine.edu

Abstract. Today, advances in short-range ad-hoc communication and mobile phone technologies allow people to engage in ad-hoc collaborations based solely on their spatial proximity. These technologies can also be useful to enable a form of timely, self-organizing emergency response. Information about emergency events such as a fire, an accident or a toxic spill is most relevant to the people located nearby, and these people are likely also the first ones to encounter such emergencies. In this paper we explore the concept of decentralized ad-hoc collaboration across a range of emergency scenarios, its feasibility, and potentially effective communication protocols. We introduce the *LocalAlert* framework, an open source agent simulation framework that we have developed to build and test various forms of decentralized ad-hoc collaboration in different emergency situations. Initial experiments identify a number of parameters that affect the likelihood of a successful response under such scenarios.

Keywords: decentralized spatial computing, decentralized ad-hoc collaboration, emergency situation management, agent framework, ad-hoc communication, ad-hoc communication protocols.

1 Motivation

Technological advances in mobile phones, location-based social network applications, and ad-hoc communication abilities will change the ways in which people respond to emergency situations in the future. Emergencies vary greatly, from far reaching events such as fast moving wild fires, hurricanes, or flooding, to events experienced on a smaller spatial scale such as a bomb threat in public building, an assailant at a school or university, or an accident at a local chemical plant. These events are characterized by occurring suddenly, requiring immediate reaction, and being first encountered by people in close proximity to the event.

In the domain of geosensor networks, the term decentralized spatial computing was coined [13] to capture the fact that while individual sensor nodes only capture a local glimpse of a geographically larger phenomenon, they can collaborate with their immediate local neighbors to identify a larger phenomenon. In this context, global control or coordination is not necessary, nor do local nodes

need to understand the global phenomenon to coordinate locally. The paradigm of ad-hoc situational collaboration could be similarly powerful in emergency response situations that involve information related to spatio-temporal events and the people located their proximity.

Imagine the following scenario: Mary is shopping at a local superstore. While she sorts through some items on a shelf, she hears a person screaming and a shot being fired. She can determine the general direction of the sound, but she cannot find out what is really happening. Alarmed and scared, Mary checks her smartphone application, *LocalAlert*, a real-time location-based spatial event notification and coordination application. *LocalAlert* is location-aware, and enables short-range communication between people in spatial proximity without the need for users to know each other or connect to a centralized infrastructure. *LocalAlert* recognizes Marys location and can detect other people in her proximity. The application might then display (via text or graphics) information about the already-known event in the store. If not, Mary can ask a question that is forwarded to others in her proximity and ultimately relayed to people who may have encountered the event first hand. Mary is now better informed and decides to leave the store using a safe route. Mary continuously checks *LocalAlert* for updates in case the shooter has moved, and evaluates various escape routes. Using *LocalAlert*, Mary can retrieve up-to-date information about the situation as provided by other people in the same emergency. Other scenarios include a bomb threat in a public building or apartment complex, or a fast moving wildfire. *LocalAlert* can help to identify shortest evacuation routes for people unfamiliar with a building floor plan, or display notifications about blocked routes by other people who encountered them. *LocalAlert* is not restricted to emergency situations either; it can also be useful in other location-based proximity scenarios. For example, drivers might be stuck in a suddenly occurring traffic jam and want to know the length of the traffic jam or its cause.

Today, several technical and non-technical challenges remain. While GPS can be used for outdoor localization, determining accurate indoor location is still an active research area; this is relevant if *LocalAlert* is combined with mobile mapping. Further, todays mobile phones and smartphones have limited ad-hoc communication abilities based on short-range radio devices and ZigBee-based mesh networks [21]. This, how-ever this is changing rapidly due to the advantages of secure short-range communication enabled by ZigBee. User interface questions also remain: in which form should information about events be created? It might take too long to type in textual event messages in time-critical emergency situations, and text-based messages are difficult to automatically aggregate.

Decentralized self-organizing applications for emergency situations do not (yet) exist. Our first objective is to test the general feasibility of such an approach. Our second objective is to investigate different ad-hoc communication protocols and coordination strategies between smart device users to identify effective protocols under different circumstances. For example, if users already have partial event information, they might prefer to pull (*ask*) for additional information. On the other hand, people first discovering a suspicious event will

likely alert (*tell*) other, unaware people in their vicinity who could be affected. Or perhaps some combinations of the two approaches would occur? Which communication protocol leads to information saturation in the system quicker? What are the key parameters of such an information dissemination system, and how do these parameters depend on each other? Beside decentralized *notification* of an event, can *collaborative coordination* also be achieved? To investigate the feasibility and limitations of ad-hoc decentralized coordination we have implemented the *LocalAlert* simulation framework. In *LocalAlert*, smartphone users are modeled as agents in a spatial environment in which they follow routes and accomplish objectives. The simulation environment accommodates both indoor and outdoor spaces, a rich set of dynamic event types, and a range of ad-hoc communication protocols and coordination strategies. We tested them under varying input conditions, such as different agent populations, event types, and behaviors, and tested the efficiency of the notification and coordination strategies.

The remainder of this article is structured as follows. In Section 2, we describe the technological background of this research to demonstrate feasibility. Section 3 introduces our *LocalAlert* simulation environment and Section 4 contains our performance analysis. Section 5 discusses related work, and Section 6 offers our conclusions and identifies possible future work.

2 Background

In this section, we present the background consisting of different research areas and technologies that are related to our exploratory approach. We also briefly discuss the state of the art in these areas to assess the feasibility of our approach.

2.1 Ad-Hoc Communication Technology

Ad-hoc communication technology [7] has been available for several decades and has found widespread application in *wireless sensor networks*, *mesh networks* and *mobile ad-hoc communication networks* (MANETs). Instead of relying on preexisting routing infrastructure with routers or access points, a wireless ad-hoc communication network is decentralized. Here, all network devices have equal status and can connect with any other devices in their wireless link range. The communication topology of the network is built *ad-hoc* based on node proximity, availability, and wireless link properties, and devices participate in the *routing* of messages by forwarding data to other more distant nodes via multiple “hops” (see Figure 1). The routing methods in ad-hoc networks attempt to *dynamically* find paths between two nodes A and B. The presence of dynamic and adaptive routing protocols make it possible to set-up ad-hoc networks quickly, with minimal configuration, and enable dynamic restructuring on-the-fly since the devices do not need to be known by name ahead of time. This fact makes them well suited for use in a wide range of emergency situations, including natural disasters or military conflicts.

Today, most mobile phones support several types of wireless communication, such as communication over cellular, Wi-Fi and Bluetooth networks. Efforts to

provide built-in support for ad-hoc networking in mobile phones are also taking place. For example, ZigBee [21], a widely used interoperability standard specification for various ad-hoc networking protocols, includes the *ZigBee Telecom Services* standard [22] for value-added services such as mobile gaming, secure mobile payments, and mobile advertising. Also, the ZSIM card has been proposed, which provides local ad-hoc communication using the ZigBee mesh protocol and supports local ad-hoc links over distances up to 70m indoors and 400m outdoors. Ad-hoc communication using mobile phones, however, should not be confused with cell broadcast [6] for GSM-based mobile phones. Cell Broadcast (CB) messaging is a one-to-many, geographically focused messaging service that allows users to broadcast a text message anonymously and simultaneously to all phone subscribers currently located within a cell of the larger cellular network. This service however is not available to the average subscriber.

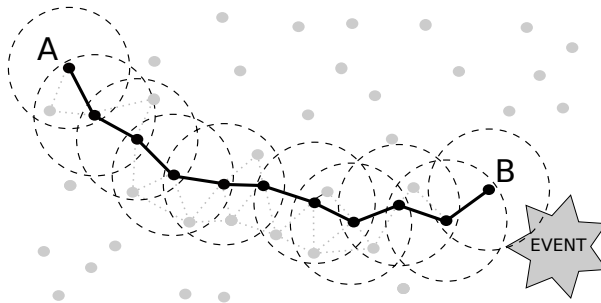


Fig. 1. Ad-hoc communication

2.2 Ad-Hoc Collaboration

Ad-hoc communication enables ad-hoc collaboration between computing nodes. It has been introduced as a robust and flexible paradigm by wireless sensor networks to cooperatively accomplish tasks [5]. Ad-hoc collaboration is a higher-level concept than ad-hoc communication; it can be defined as the collaboration of several computing nodes located in spatial proximity to achieve a task, even in the absence of previous communication or collaboration. Since there is no central coordinator with global knowledge that assigns roles or partial tasks to the nodes, every node can act to initiate collaboration and decide to participate in collaboration initiated by other nodes. This mode of collaboration is also called decentralized collaboration. In geosensor networks, ad-hoc collaboration has been used to aggregate locally sensed information collaboratively into global knowledge about a phenomenon such as establishing its currently estimated boundary [8,9]. In mobile geosensor networks, sensor nodes participate in ad-hoc collaboration with nodes that they encounter in their spatial proximity while moving, and then pass on information to them. Examples of this include vehicles communicating about hazardous road conditions [14] or information exchanges in intelligent transportation networks [16].

2.3 Smartphones and Emergency Management

Today, smartphones account for about a third of the mobile phone market [12]. They are often equipped with GPS, and enable location-based social network applications such as *Foursquare*, *Gowalla*, and *Google Latitude*, which allow users to “check in” to places in real-time. Other location-based social applications include mobile friend finders, mobile gaming applications, and dating applications. Although today, people use social media applications on smartphones to share their location (and potentially other) information in real-time, ad-hoc communication based applications for ad-hoc collaboration do not (yet) exist.

Smartphones and similar mobile devices are also currently used for emergency management. Applications exist that let people store “in-case-of emergency” data on their smartphone – such as critical contact information, a list of current health care providers, or severe allergies – for easy access. Additionally, smartphones and similar devices are used as platforms for centralized updates about emergencies by cities and states. For example, Cupertino, California launched an emergency application that acts like a Rolodex with critical information in case of an emergency (such as earthquakes, wildfires, etc.) with real-time weather and hazard alerts, as well as with meeting place and shelter locations [3,10]. Similar applications are available in other cities.

3 Simulating Agent-Based Decentralized Ad-Hoc Collaboration in Emergency Situations

In this section, we describe the important components of the *LocalAlert* simulation framework, and specify the problem space we have investigated, implemented and tested. To enable modeling of ad-hoc collaboration in emergency situations we conceptualized different types of *space*, *agents* and spatial emergencies (called *events*). Furthermore, we modeled *ad-hoc collaboration strategies* between agents in detail. Our objectives are to firstly investigate the feasibility of this approach and secondly, to identify which collaboration protocols work well under which circumstances.

3.1 Modeling Space and Events in *LocalAlert*

In the *LocalAlert* framework the space serves as a shared environment for all agent entities.

Shared, dynamic space: We provide a base space, represented by an adjustably sized two-dimensional grid of cells, on which entities like agents and events exist. The space is configurable as a combination of freely navigable spaces and obstacles, thus, supporting the modeling of a wide range of indoor or outdoor spaces of varying complexity. The space is composed of patches, which are either *non-agent-barrier* patches, which act as freely navigable space for agents, or *agent-barrier* patches, which represent cells an agent may not travel over (e.g. event physical barriers and exits). All patches

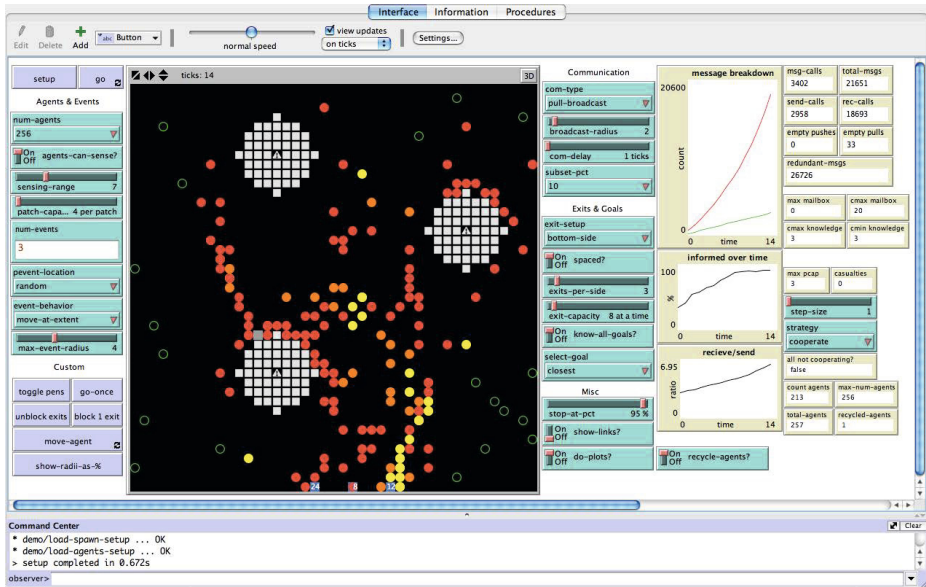


Fig. 2. The *LocalAlert* framework simulating agents and events

have a patch-capacity. The *LocalAlert* framework supports the use of more complex base-maps; however, they are not used in our current analysis.

Events: Events are defined as an additional but separate agent entity class. Events possess attributes and follow rules which determine their overall behavior: since we are modeling dynamic spatio-temporal events, these behaviors include properties such as maximum expansion radius and rules which dictates how moving events act when they reach a wall. For simplicity, agent-event interaction is limited by the following rules:

1. Agents may freely and safely observe events at any distance greater than one patch. This helps agents in the space who are trying to maintain both their intended heading and some desired buffer distance from all proximate events.
2. Agents and events cannot safely occupy the same location in space at the same time, and agent safety is currently a binary property: completely unharmed (safe), or completely injured (consumed by the event). This reduces model and framework complexity while still producing viable data as it relates to the effectiveness of a given strategy.

3.2 Modeling Agents in *LocalAlert*

Agent entities are modeled around mobile smart-device users, who all have the common goal of obtaining and then disseminating information about spatio-temporal events occurring in the shared space. Agents can sense events directly based on a sensing-range parameter, or via the exchange of information with

other agents in the space (limited by a communication-range parameter). They distinguish between two objectives: either wandering freely or responding to an event, depending on the types and amount of information they currently possess. In the default wandering state, agents move freely around the currently unconstrained space and retain this state until they either decide to exit on their own, or encounter or are informed of an event. In either case, when agents switch into the second state of responding, their primary objective becomes quickly but safely exiting the space. The range of an agent's abilities to communicate and sense is determined primarily by the framework parameters. Furthermore, in case of encountering an event first-hand, agents are tasked to perform the collective process of decentralized emergency coordination and self-organization.

3.3 Ad-Hoc Communication and Collaboration Strategies in *LocalAlert*

Since communication is central to many of the questions related to this research, *LocalAlert* features an extensive set of communication protocols that determine how agents can communicate with each other. Communication involves agents exchanging messages that contain a variety of spatial information about events and the space it-self. In the simplest form, a message consists of a unique message ID, a message body, and a location, which refers to the location of the spatial entity that the message refers to (i.e. the event). Thus, updates about event locations can be accommodated in this model. Additional message fields, such as the number of hops a message has taken before being received, are also maintained.

Communication: We have currently implemented two communication types: *push* (agents send information) and *pull* (agents request information) strategies. Additionally, for each communication type we investigate two message distribution models, *epidemic* and *broadcast*. Under the epidemic message distribution model, the number of other nodes selected for communication varies from 10% to 90% of the available neighbors in communication range, while in the broadcast mode, a message is flooded across the network (i.e. each agent receiving a message forwards it to *all* other agents in its own communication range).

Collaboration: Collaboration consists of *notification* and *adaptive coordination*. In the notification mode, agents simply request information about an event or inform others about an event. In the collaborative mode, agents exchange spatial information about the *space* (e.g. which exits are blocked?) and the *event* (does the event change location? Where is it now?). Both modes are forms of dynamic collaboration, where old messages about the same event can be updated with new information. We compare three different levels of agent collaboration:

1. *Sensing only:* in this mode, agents do not cooperate or communicate with other agents in the space in any way; all information is obtained solely through an agents sensory capabilities. Thus, the agent has to encounter the event. This basically reflects todays situation, where ad-hoc communication enabled smartphones are not used.

2. *Selfish*: in the selfish mode, agents collaborate only until they have fulfilled personal exit requirements, at which point they no longer actively participate in communication with other agents, though they may still act as intermediate nodes in forwarding information to others.
3. *Cooperate*: agents collaborate fully with others for the full duration of their time in the space.

Agent communication is modulated by several framework parameters, i.e. how often agents communicate, with how many other agents an agent communicates, and how many messages an agent may store.

During the communication and collaboration processes, agents perform intelligent message aggregation. Currently, agents combine messages based on the identification of an existing message with matching *location* and *message body* fields. This, along with the other various message fields, allows agents to rank information in a number of ways: for example, an agent may sort all known exit locations by proximity, or by the number of times the agent has received a message using the *times-heard* field. The *LocalAlert* framework also provides additional message handling and decision-making support mechanisms for an agent, such as managing a blacklist, which contains messages that are no longer suitable to pass on, e.g. messages about a previously known safe exit, which is then discovered to be blocked. Upon discovering any such invalid information, agents purge all matching messages from their current message and knowledge lists, with the intent of ensuring out-of-date information is no longer spread.

3.4 Implementation

The *LocalAlert* framework is implemented in NetLogo [19], a free, cross-platform, programmable multi-agent modeling environment. NetLogo is particularly useful for the investigation of models that have dominant spatial or temporal elements, or systems models, which evolve over time. Our strategy was to encapsulate the newly developed core functionality into small, purpose-built modules so that the *LocalAlert* framework is extensible, reusable, and easily expandable to accommodate new functionality without changes to existing code. The code is available at <http://code.google.com/p/gaem/>.

4 Performance Evaluation

4.1 Test Parameters

Our interest is in identifying optimal communication and coordination strategies under a variety of emergency situation scenarios. We constructed a series of nine experiments, representing different combinations of event and response components. These experiments are designed to investigate the influence of the following parameters on our proposed response models: *agent population* or density (256 vs. 512 vs. 1024 agents), *coordination strategy* (cooperate, selfish, or sensing only), *communication protocol* (push vs. pull-based, broadcast vs. epidemic),

event type (single fixed, single expanding, single moving, multiple events), and initial *event location* with regard to exit locations. Results from each run are ranked according to the metric “*ticks-to-completion*” which represents the number of iterations required to reach an exit criterion. Ticks to completion serves as the most telling measure of effectiveness since the faster agents are informed, the faster they can make informed decisions and exit. However, as time is not the only measure of effectiveness, we also examine the robustness of a response strategy, as it relates to how likely the strategy is to succeed.¹

4.2 Validating Decentralized Ad-Hoc Collaboration

Table 1 shows the results for an *expanding* event scenario. We evaluated two criteria: *speed* (minimum ticks-to-completion) and *reliability* (how likely a given strategy is to succeed). For each agent population, a total of four columns are presented: the first two columns represent the top 10th percentile of successful runs (raw count and percentage), and 3rd and 4th column represent the number of successful runs per strategy and its percentage of the overall runs. For example, we simulated 720 runs with 256 agents, and 188 of the successful runs used the full cooperation strategy, while 170 runs used either the selfish or sensing only strategies. Additionally, for each agent population, a pass rate (PR) value is provided, representing the total pass rate (number of successful runs/number of runs tested) regardless of configuration: for example, roughly 73% in the case of an expanding event with 256 agents. For expanding events, the tests show that the full cooperation strategy results in the fastest exiting of agents, with this strategy representing 60-76% of the fastest successful runs. The relative location of the event also matters – events near the exits (bottom-half) delay escaping

Table 1. Testing different collaboration strategies for expanding events

expanding	256 (73.33% PR)		512 (85.56% PR)		1024 (90.00% PR)	
	top (%)	total (%)	top (%)	total (%)	top (%)	total (%)
strategy						
cooperate	41 75.93	188 35.61	48 70.59	206 33.44	59 60.82	216 33.33
selfish	13 24.07	170 32.20	20 29.41	209 33.93	29 29.90	210 32.41
sensing only	0 0.00	170 32.20	0 0.00	201 32.63	9 9.28	222 34.26
location						
bottom-half	0 0.00	187 35.42	0 0.00	223 36.20	56 57.73	204 31.48
center	35 64.81	236 44.70	33 48.53	215 34.90	29 29.90	240 37.04
top-half	19 35.19	105 19.89	35 51.47	178 28.90	12 12.37	204 31.48
com-type						
push	25 46.30	259 49.05	20 29.41	303 49.19	38 39.18	324 50.00
pull	29 53.70	269 50.95	48 70.59	313 50.81	59 60.82	324 50.00
subset						
10%	28 51.85	265 50.19	11 16.18	333 54.06	25 25.77	345 53.24
100%	26 48.15	263 49.81	57 83.82	283 45.94	72 74.23	303 46.76

¹ Our successful run condition is that 95% of the agent population safely exited.

Table 2. Testing different collaboration strategies for moving events

moving	256 (57.64% PR)		512 (79.17% PR)		1024 (87.64% PR)	
	top (%)	total (%)	top (%)	total (%)	top (%)	total (%)
strategy						
cooperate	28 65.12	181 43.61	41 67.21	213 37.37	46 66.67	212 33.60
selfish	15 34.88	128 30.84	20 32.79	189 33.16	18 26.09	206 32.65
sensing only	0 0.00	106 25.54	0 0.00	168 29.47	5 7.25	213 33.76
location						
bottom-half	3 6.98	161 38.80	2 3.28	210 36.84	36 52.17	195 30.90
center	19 44.19	156 37.59	22 36.07	178 31.23	31 44.93	230 36.45
top-half	21 48.84	98 23.61	37 60.66	182 31.93	2 2.90	206 32.65
com-type						
push	16 37.21	195 46.99	21 34.43	270 47.37	29 42.03	312 49.45
pull	27 62.79	220 53.01	40 65.57	300 52.63	40 57.97	319 50.55
subset						
10%	21 48.84	206 49.64	8 13.11	302 52.98	8 11.59	323 51.19
100%	22 51.16	209 50.36	53 86.89	268 47.02	61 88.41	308 48.81

agents, and reduce the chance that distant agents will learn of the event. We can also see in the fastest cases that a pull strategy outperforms a push strategy; however, they are roughly equal overall. Similarly, an epidemic messaging strategy with a 10% forwarding rate is just as effective as flooding for low density populations; however, a flooding-based strategy leads to (not surprisingly) faster success with larger populations due to more rapid information saturation. Again, when looking at all successful runs, both are similarly effective. Minimizing messages is not necessarily a concern in this setting, but it might be practically relevant that these communication strategies also work if not all agents participate.

Table 2 shows the results for *moving* events, which can obstruct exits and disturb agents' exit routes. As we can see in Table 2 (similar to Table 1), low density populations are less likely to achieve high overall success compared to higher agent densities.

4.3 Evaluation of Different Decentralized Coordination Strategies

Figures 3-5 capture the numbers of agents informed over time based on different collaboration strategies, types of events and agent populations (different color lines for each density, solid for push and dotted for pull). Figure 3 shows a stationary event located in the center of the space. As can be seen, a cooperative strategy leads to (for the highest agent density, in purple) nearly 70% of agents being informed quickly and then exiting the space quickly (around 90 ticks). Under the selfish and sensing only strategies, agents remain in the space until nearly everyone is informed or has encountered the event first hand. Figure 4 depicts an event centered in the space that expands over time. Due to the dynamic changes of the event, agents are informed quickly, but spend more time in the space due to the need to adapt and 'replan' their exit route, slowing their exit process. Figure 5 presents the results of a moving event, which show similar results compared to an expanding event.

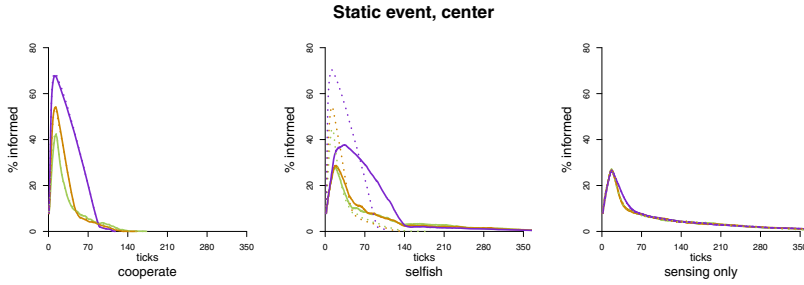


Fig. 3. Informed agents over time with a stationary event in the center

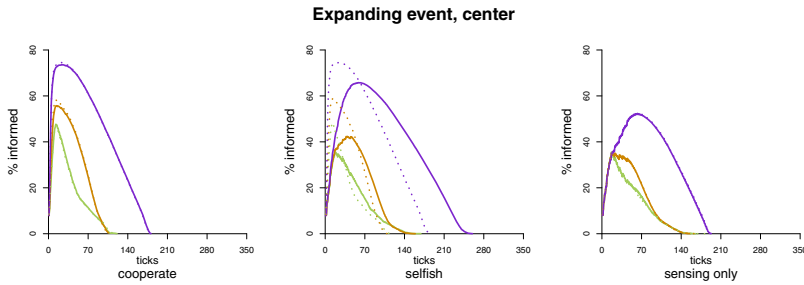


Fig. 4. Informed agents over time with an expanding event in the center

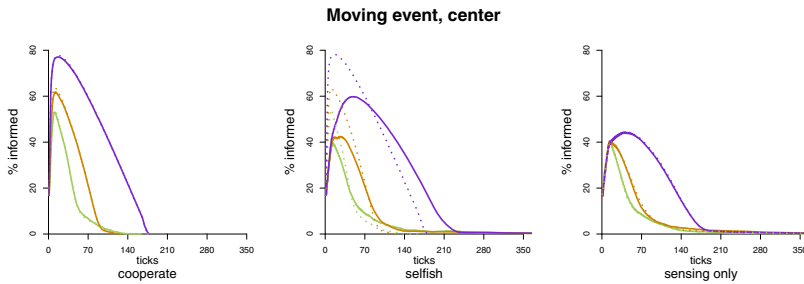


Fig. 5. Informed agents over time with a moving event in the center

In summary, for nearly all scenarios examined, the cooperative agent strategies far outperform their sensing only (uncooperative) or selfish counterparts. Under cooperative cases, initial event-related information dissemination occurs more quickly, and across a larger percentage of the total population than under uncooperative and selfish models, and as a result, a higher percentage of the total agent population was able to successfully exit, more quickly. The goodness of communication strategies (push or pull; epidemic or broadcast routing protocols) varies slightly depending on the specific event type and initial location; but there is no clear or consistently better selection. Results do however indicate that event type and location (relative to exits or goals) plays a significant role in the emergent agent behaviors over time.

5 Related Work

Today, several simulation tools exist that facilitate the investigation of various aspects of social agents collaborating in spatio-temporal environments [1,2,4,11]. While many of these tools have gained widespread publicity, there is currently no single simulation environment that allows practical investigation of all the components (technical, social, and environmental) relevant to our proposed application.

The idea of decentralized ad-hoc collaboration has been successfully established in wireless sensor networks and especially in geosensor networks, in which the concept of location, local events and node proximity to spatial events is poignant. Ad-hoc collaboration has also been used to aggregate local sensor information to form knowledge about global event such as establishing and tracking event boundaries [8,9]. In *mobile* geosensor networks, sensor nodes use ad-hoc collaboration with nodes they encounter in spatial proximity to pass on information about e.g. hazardous road conditions [14], exchange information about potential rideshares in intelligent transportation networks [16], or collaborate on capturing currents in ocean sensor networks [15].

While [17,18] explore ad-hoc collaborative decision making in spatio-temporal environments; this work focuses on complex collaborative tasks such as toxic spill clean-up and agents with varying abilities. Our framework could be useful for exploring collaboration strategies for more complex tasks in emergency situation such as rescuing victims or directing crowds through a space that is unknown to most participants.

[20] explores an idea that is similar to ours as presented in this paper. However, this work focuses more on the representation and sharing of partial spatial knowledge and creating ad-hoc local maps of a graph/map structured outdoor environment using only a broadcast strategy, while our work investigates several different communication protocols (push vs. pull, and broadcast vs. epidemic) and also explores various types of events (e.g. moving events). We also propose the *LocalAlert* simulation environment as the basis for more exploration under this research question. Overall, both approaches come to similar conclusions; mainly that ad-hoc collaboration enables a better outcome in emergency situations.

6 Conclusions

In this paper, we investigated the potential of smartphone based ad-hoc collaboration in emergency situations. We presented the *LocalAlert* simulation framework, designed to simulate various ad-hoc collaboration protocols for agents dynamically reacting to spatio-temporal events. We tested agents acting alone (sensing only), selfishly, and under a fully cooperatively behavior model, and our results indicate that such an application is indeed valuable. Under cooperative models, information dissemination occurred most quickly over the largest percentage of the population, and as a result, a greater percentage of the total population was able to successfully exit in less total time. This paper serves

as a first exploratory analysis of several possible and likely communication and coordination strategies.

In the future, the *LocalAlert* framework will be used to perform a much more in-depth analysis. We also make the *LocalAlert* framework available as open source code so that it is available to the community for continued development of new modules and to introduce other options, such as additional communication models, agent social behaviors, or spatial layouts. There are still many open, interesting research questions related to this work, which need to be addressed in other research areas of GIScience. For example, which human user interface is most appropriate for such an application? What is the best way to represent imprecise spatial knowledge and support automatic reasoning about it? How can we aggregate imprecise spatial knowledge from different sources automatically? The authors of this paper plan to continue exploring such interdisciplinary questions, and hope that this work serves to encourage other GIScience researchers to do the same, so that a real-world implementation of the *LocalAlert* framework may one day exist.

References

1. Buzing, P.C., Eiben, A.E., Schut, M.C.: Evolving Agent Societies with VUScape. In: Banzhaf, W., Ziegler, J., Christaller, T., Dittrich, P., Kim, J.T. (eds.) ECAL 2003. LNCS (LNAI), vol. 2801, pp. 434–441. Springer, Heidelberg (2003)
2. CASOS: Construct, <http://www.casos.cs.cmu.edu/projects/construct/>
3. City of Cupertino: Ready 95014 iPhone/iPad App, <http://www.cupertino.org/index.aspx?page=87>
4. Epstein, J., Axtell, R.: Growing Artificial Societies: Social Science from the Bottom Up. The MIT Press (1996)
5. Estrin, D., Govindan, R., Heidemann, J., Kumar, S.: Next century challenges: scalable coordination in sensor networks. In: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom 1999, pp. 263–270. ACM, New York (1999)
6. European Telecommunications Standards Institute EMTel: Analysis of the Short Message Service (SMS) and Cell Broadcast Service (CBS) for Emergency Messaging applications. Tech. Rep. 102 444 V1.1.1, ETSI (February 2006)
7. Feeney, L.: A Taxonomy for Routing Protocols in Mobile Ad Hoc Networks. SICS Research Report (1999)
8. Jiang, J., Worboys, M., Nittel, S.: Qualitative change detection using sensor networks based on connectivity information. *GeoInformatica* 15, 305–328 (2011), <http://dx.doi.org/10.1007/s10707-009-0097-0>, doi:10.1007/s10707-009-0097-0
9. Jin, G., Nittel, S.: Efficient tracking of 2d objects with spatiotemporal properties in wireless sensor networks. *Distributed and Parallel Databases* 29, 3–30 (2011), <http://dx.doi.org/10.1007/s10619-010-7075-2>, doi:10.1007/s10619-010-7075-2
10. Mercury News: City of Cupertino launches emergency app for iPhone, iPad, <http://iphoneapps-review.net/city-of-cupertino-launches-emergency-app-for-iphone-ipad/>

11. Moon, I.C., Carley, K.M.: Self-organizing social and spatial networks under what-if scenarios. In: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2007, pp. 252:1–252:8. ACM, New York (2007), <http://doi.acm.org/10.1145/1329125.1329430>
12. Nielsen: NielsenWire, Nieslen Blog, <http://blog.nielsen.com/nielsenwire/?p=28237>
13. Nittel, S.: A survey of geosensor networks: Advances in dynamic environmental monitoring. *Sensors* 9(7), 5664–5678 (2009), <http://www.mdpi.com/1424-8220/9/7/5664>
14. Nittel, S., Duckham, M., Kulik, L.: Information Dissemination in Mobile Ad-Hoc Geosensor Networks. In: Egenhofer, M. J., Freksa, C., Miller, H.J. (eds.) *GIScience 2004*. LNCS, vol. 3234, pp. 206–222. Springer, Heidelberg (2004)
15. Nittel, S., Trigoni, N., Ferentinos, K., Neville, F., Nural, A., Pettigrew, N.: A drift-tolerant model for data management in ocean sensor networks. In: Proceedings of the 6th ACM International Workshop on Data Engineering for Wireless and Mobile Access, *MobiDE 2007*, pp. 49–58. ACM, New York (2007), <http://doi.acm.org/10.1145/1254850.1254860>
16. Nittel, S., Winter, S., Nural, A., Cao, T.: Shared ride trip planning with geosensor networks. In: Miller, H.J. (ed.) *Societies and Cities in the Age of Instant Access*, *GeoJournal Library*, vol. 88, pp. 179–194. Springer, Netherlands (2007)
17. Raubal, M., Winter, S.: A spatio-temporal model towards ad-hoc collaborative decision-making. In: Painho, M., Santos, M.Y., Pundt, H. (eds.) *Geospatial Thinking*. Lecture Notes in Geoinformation and Cartography, pp. 279–297. Springer, Heidelberg (2010)
18. Raubal, M., Winter, S., Dorr, C.: Decentralized Time Geography for Ad-Hoc Collaborative Planning. In: Hornsby, K.S., Claramunt, C., Denis, M., Ligozat, G. (eds.) *COSIT 2009*. LNCS, vol. 5756, pp. 436–452. Springer, Heidelberg (2009)
19. Wilensky, U.: *NetLogo*, Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL (1999), <http://ccl.northwestern.edu/netlogo>
20. Winter, S., Richter, K.F., Shi, M., Gan, H.S.: Get me out of here: collaborative evacuation based on local knowledge. In: Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, *ISA 2011*, pp. 35–42. ACM, New York (2011), <http://doi.acm.org/10.1145/2077357.2077365>
21. ZigBee Alliance: ZigBee, <http://zigbee.org/>
22. ZigBee Alliance: ZigBee Telecom, <http://zigbee.org/Standards/ZigbeeTelecomServices/Overview.aspx>

High-Level Event Detection in Spatially Distributed Time Series

Avinash Rude and Kate Beard

School of Computing and Information Science, University of Maine, Orono, Maine
beard@spatial.maine.edu, avinashrude@gmail.com

Abstract. This paper presents an approach for the detection of high-level events from spatially distributed time series. The objective is to detect spatially evolving high-level events as aggregate patterns of primitive events. The approach starts with a segmentation of time series into primitive events as building blocks for high-level events. A high-level event ontology is then used to specify the composition of high-level events of interest in terms of initiating, body forming, and terminating primitive events. We illustrate the approach first with simulated time series data to identify traffic congestion events and then with real data to identify storm events from sensor time series collected as part of an ocean observing system deployed in the Gulf of Maine. Detected storm events are compared against NCDC reported storm events as an evaluation of the approach.

Keywords: event detection, time series segmentation, primitive event.

1 Introduction

With the expansion of sensor monitoring systems and particularly wireless sensor networks (WSN), data in the form of spatially distributed time series are becoming increasingly common. The configuration assumes a set of fixed or mobile sensor nodes or other data observation platforms deployed within a region and generating time series on one or more variables. Applications for such data include spatial temporal interpolation of dynamic fields [1,2], detection of topological changes in an evolving field [3], event detection [4, 5], and event tracking [6,7]. This paper reports on an approach for identifying high-level events from sets of spatially distributed time series observed at fixed locations within a region. A high-level event in this context refers to a complex, often multivariate phenomenon with spatial properties (e.g. spatial extent, movement) that evolves over time such as a storm, forest fire, disease outbreak, industrial accident, or traffic jam. Extraction of interesting occurrences from time series has been a focus of research in many disciplines [5,6,7,8,9], with work on detection of events in both univariate [10] and multivariate time series [11,12,13,14,15]. A gap exists, however, between low level events detectable by sensors and high-level events recognized by humans. A single sensor time series at a location typically sees partial evidence of a high-level event but often not a complete picture [13].

High-level events may be conceptualized in a gestalt view, where the whole pattern of a physical, biological, or psychological phenomena, is integrated so as to constitute a functional unit with properties not derivable from its parts. Alternatively, a partitive view conceives of a high-level event as composed of constituent parts that contribute to characterization of the phenomena. This paper assumes a partitive view, in which high-level events are configurable from component parts (spatial, temporal and thematic) extracted from sensor generated univariate time series.

Different approaches have been considered for high-level event detection but usually not addressing multivariate, spatio-temporal dimensions together. Knowledge discovery methods address some aspects of the problem through time series segmentation, clustering and classification, and some methods exist to monitor time series and detect events in parallel. Machine learning approaches have analyzed multiple time series from single locations but most have not addressed the discovery of spatio-temporal features extending over multiple locations [4]. Spatio-temporal scan statistics [15] detect clusters as high-level events by searching for spatio-temporal clustering of a single event type, e.g. emergency department visits, but these methods generally do not address multivariate dimensions of a high-level event. Velipasalar et al [16] generate composite events from multiple camera based primitive events that address thematic and temporal constraints but do little with the spatial dimension. Batal et al [14] addresses multivariate time series segmentation and segment clustering for context identification but also do not focus on the spatial dimension. This paper describes a general model for high-level event detection and characterization in settings with several geolocated multivariate time series. The approach uses primitive events as units of univariate spatio-temporal change and assembles these into units of multi-variate spatio-temporal change. In other words primitive events are the spatio-temporal and thematic parts that form high-level events according to temporal, or spatio-temporal, and thematic constraints. Section 2 of the paper describes primitive events. Section 3 describes the model for high-level events as composites of primitive events and the assembly methodology. Section 4 describes the high-level event detection approach applied to simulated data. Section 5 describes a case study which applies the approach to real data for storm detection and characterization.

2 Primitive Events

The specification and detection of primitive events is the foundation for the approach. A primitive event in this context is a subsequence of a time series for which a particular property of a parameter holds, typically indicating a state or change of state over a temporal interval. There are many possible subsequences that could form primitive events depending on states of interests and high-level events to be constructed.

Given a set of time series denoted TS_s^p , indexed by parameter (p) and location (s), primitive events are generated by application of abstraction functions [18] to the time series. Abstraction functions (AF) can be threshold based [19], pattern-based [20,21], or learning based approaches [22,23,24]. An abstraction function generates an

abstraction type which is a general abstract state used to classify primitive events. Abstraction types include value types that classify a parameter's value (e.g. high, medium, low) and trend types that classify the derivative of the parameter's value with subtypes, gradient (falling, rising) and rate (rapid, moderate), corresponding respectively to the sign and magnitude of the derivative. A threshold abstraction function applied to the first derivative of a temperature time series, for example, would generate a trend abstraction type primitive event (e.g. falling temperature event). An abstraction function, in addition to generating an abstraction type, establishes a time interval for primitive events. For an individual time series $TS = \{x_k, 1 \leq k \leq N\}$ labeled by time points t_1, \dots, t_N , a primitive event $PE[ta, tb]$ is a segmentation of TS into consecutive observations x_a, x_{a+1}, \dots, x_b , $a \leq k \leq b$ where x_k satisfies conditions of the AF. As diagramed in Fig. 1, for a set of spatial locations $S = \{s_1, \dots, s_j\}$ and one to m time series, $TS_{s_j}^p$ $p=1, \dots, m$, for each location, application of an AF generates primitive events $PE_{s_j}^p[ta, tb]$ that are classified by the parameter(s) p and abstraction type (AT). The minimum attributes for a primitive event are thus: a unique event ID, locationID, event type formulated from parameter p and abstraction type (AT) and start and end times. Abstraction functions can be designed to run in different settings, (e.g. a WSN setting at the sensor node level in near real time, or for historic sets of time series in a database context).

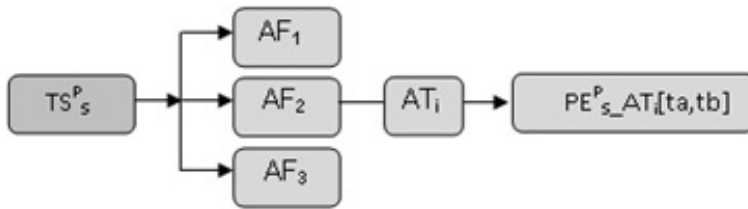


Fig. 1. Sets of abstraction functions (AF) applied to time series (TS) establish a basic abstraction type (AT) and time interval for each primitive event (PE)

3 High-Level Events Modeled from Primitive Events

Galton and Mizoguchi [25] and others argue that it is meaningful to think of events as having parts. We can think of high-level events as having thematic parts, analogous to organs of a body and temporal parts such as initiating conditions that differ from conditions signaling the termination of an event. Spatial parts of an event can describe different spatial partitions such as core versus periphery, or that part of a flood surrounding my house. Primitive events, as spatio-temporal thematic units, contribute thematic, temporal, and spatial parts to the construction of a high-level event. This section describes the general model for high-level event composition from primitive events. A high-level event is assumed to be associated with some a priori knowledge which directs the composition. This a priori knowledge directs both what types of primitive events make up a high-level event and rules for their composition. We use a high-level event ontology to define a template for high-level events as

consisting of three sets of primitive events: initiating, body, and terminating, expressed through three key properties between primitive events and a high-level event: initializes formsBodyOf, and terminates as shown in Fig 2. A high-level event is then instantiated by associating prescribed sets of primitive events through these properties. With this template, a wide range of high-level event types can be specified such as storms, droughts, floods, forest fires, or traffic jams.

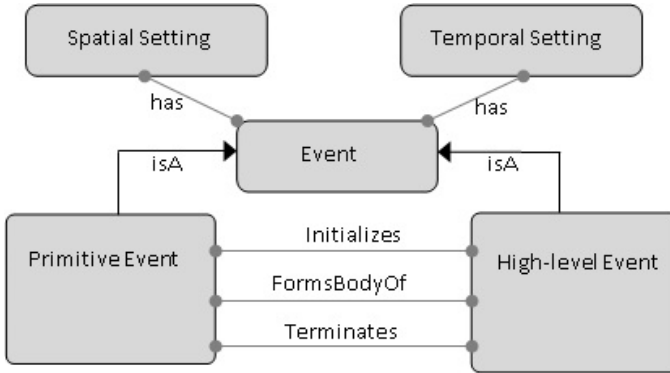


Fig. 2. Diagram of high-level event ontology indicating key properties between primitive and high-level events

Initiating primitive events are generated from time series of sensor-detectable parameters with known association to conditions for initiation of a high-level event. We refer to the sensor measurable parameters that signal the onset of the high-level event as marker parameters. Marker parameters that identify initiating conditions often identify the terminating condition as well (i.e. the recovery of a marker parameter to a normal or steady state condition often signals a termination of the high-level event). River flooding events, for example, may be recognized by ‘rapidly increasing water-levels’ exceeding a threshold and terminated by ‘recovery to normal water-levels’. More than one type of primitive event may initiate or terminate a high-level event and the body of a high-level event can be composed of any number of primitive events of different types that contribute to thematic characterization of the high level event. The body of a storm for example may be characterized by precipitation events, high wind events, or changing wind direction events, all obtainable as primitive events from sensor time series. Primitive events participating in initiating, body, and terminating relationships are referred to respectively as I, B, and T sets with respective elements I-event, B-event and T-event. The ordering of these subsets of primitive events for a high-level event are specified using Allen’s interval relations [26]. Possible interval relations between an I-event and B-event are: I-events may start, overlap, or meet B-events. B-events may finish, overlap or meet T-events. I-events must come before, overlap, or meet T-events.

We expect some influence of a high level event over a region to generate spatio-temporal correlations in the marker parameters that lead to spatio-temporal clustering in both I events and T events (i.e. a high level event is likely to be expressed at a single location in consecutive time slices as well as at nearby locations at consecutive time slices). To discover candidate high-level events, we use hierarchical clustering to identify potential initiating clusters, I_C^p of I-events and terminating clusters, T_C^p of T-events. Lin et al [27] address the idea of capturing process similarity in a clustering approach. Since primitive events represent a state change, the change or process similarity is captured in our approach by clustering on primitive events. Hierarchical clustering converges to a single cluster as illustrated by the dendrogram in Fig. 3a which shows the results for an example with four candidate I-event clusters, one of which is just a singleton I-event.

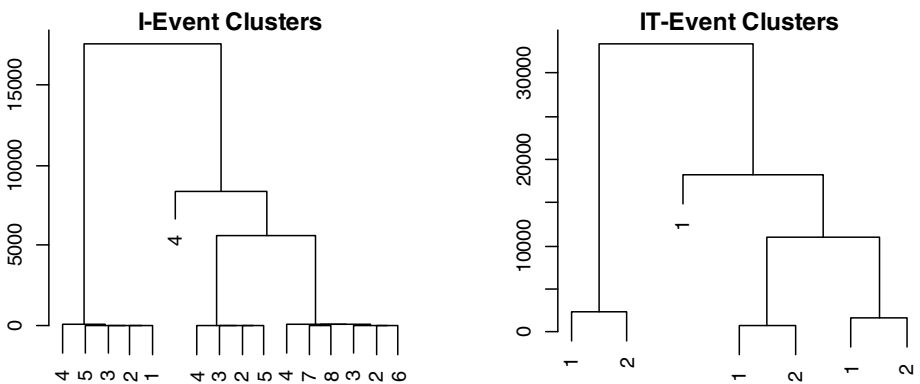


Fig. 3. Dendrogram of clustered I-events and 3b clustered I (1) and T(2) event temporal means which indicate candidate high-level events

To discover significant I_C^p and T_C^p sub-clusters, we use an inconsistency metric [28] where the height of a link is compared with link heights at the next lowest level. Significantly different link heights indicate cluster boundaries. Each I_C^p should include all the locations that “see” an initiating primitive event within a short lag time, and similarly for each T_C^p . Extracted $I_C^p[ta,tb]$ and $T_C^p[ta,tb]$ clusters obtain time intervals from constitute PE intervals where a is the minimum and b the maximum time stamp of PEs in the cluster. For each I and T cluster we obtain a temporal mean $I_C^{p,tm}$, $T_C^{p,tm}$, and a second hierarchical clustering is performed using $I_C^{p,tm}, T_C^{p,tm}$ to identify sets of paired initiating and terminating clusters which indicate starts and ends for candidate high level events. The dendrogram in 3b illustrates the results from clustering the $I_C^{p,tm}, T_C^{p,tm}$. Clusters containing paired I and T-events in this hierarchical clustering identify candidate high level events HLE_C .

Each HLE_C obtains a temporal extent from the minimum timestamp of its I_C^p and the maximum timestamp of its T_C^p . Spatial properties of candidate high-level events are developed from the spatial locations S of the constituent primitive events. As a candidate high-level event evolves within or moves across a region, a spatial pattern

is revealed through the spatio-temporal ordering of I-events and T-events at some temporal lag. We capture this pattern in a Spatial Progression String (SPS) which represents the order in which locations S detect the initiating and terminating primitive events in the I_C^P and T_C^P sets. An SPS consists of a sensor node or LocationID combined with an initiating (I) or terminating (T) flag, temporally indexed with the start times of primitive events in an HLE_C . Several SPS configurations are possible as illustrated for an example grid of sensor node locations shown in Fig. 4. The SPS pattern in 4a encodes a high-level event moving as a front from west to east (e.g. I-events register at nodes 1,2, and 3 at similar times, then at nodes 4,5,6, at similar times followed by similar times at nodes 7,8,9. T-events follow in the same order. The SPS in 4b encodes a smaller high-level event (smaller in that fewer nodes see an I-event or T-event in a time slice) tracking diagonally from southwest to northeast. The SPS in 4c encodes a high-level event moving south to north. The SPS construct is domain-independent and can be applied for the analysis of any type of high-level event in sensor network settings. For a WSN setting, the clustering steps and SPS generation would be carried out at a cluster head or base station.

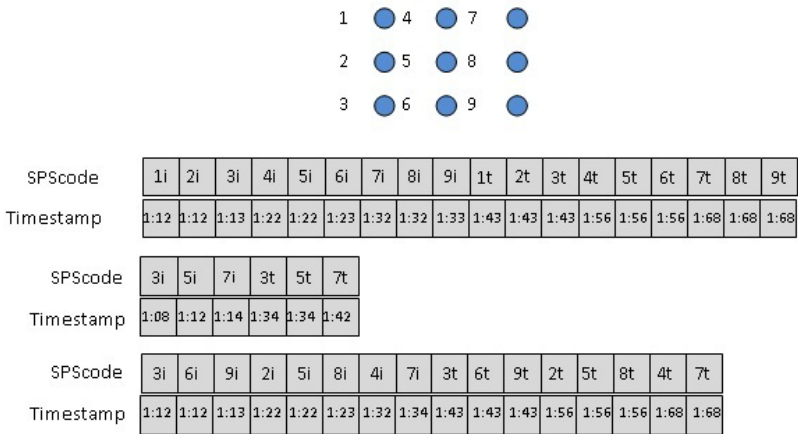


Fig. 4. Example Spatial Progression Strings (SPS) for three different spatial patterns over the set of grid locations 1-9, a) example high level event moving west to east as a front, b) event moves diagonally southwest to northeast and c) event moves south to north

4 High-Level Event Detection in Simulated Data

This section describes the high-level event detection approach applied to simulated data sets where traffic congestion events are the high-level events of interest. For this example, mean speed and vehicle density (veh#/km/lane) serve as marker parameters. Time series were generated for a linear set of sensor locations $S = \{s1, \dots, s8\}$ deployed along a roadway as illustrated in Fig. 5.

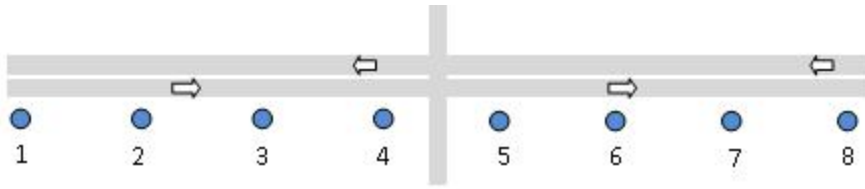


Fig. 5. Locations of sensors generating time series deployed along a roadway

A section of simulated time series is plotted in Fig. 6 showing a drop in mean speed and rise in vehicle density progressing from node 4, to 3, to 2, to 1, and losing a speed signal at 5 and 6. These marker parameters then recover in a similar order. I-events are extracted as a fall in mean speed (MS_F) and rise in vehicle density (VD_R) and the relevant T-event types correspond to a rise in mean speed (MS_R) and a fall in vehicle density (VD_F).

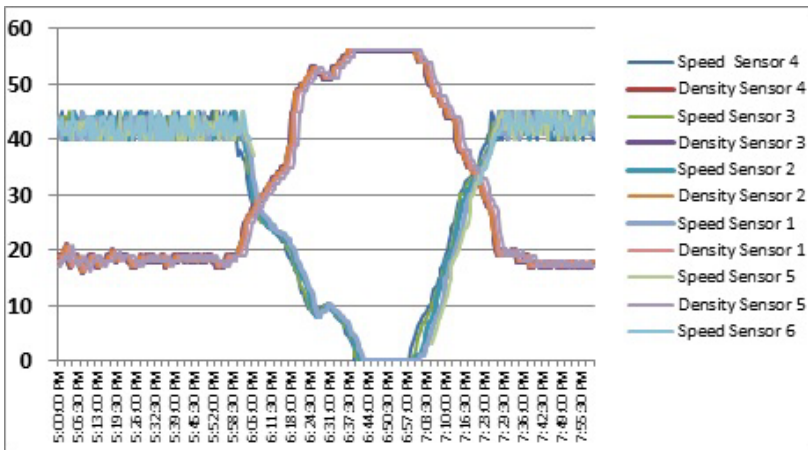


Fig. 6. Simulated time series for a high-level traffic congestion event. I-events are based on a fall in average speed and an increase in vehicle density and T-events on the recovery of these marker parameters.

A clustering of detected I-events is shown in Fig. 7a and the clustering of the I and T cluster means I_C^{tm}, T_C^{tm} is shown in 7b. The paired clusters in 7b indicate candidate high level events. To obtain spatial information for the high level events we concatenate the information in the paired initiating and terminating clusters, $I_C^{[a,b]} - T_C^{[a,b]}$ to form an SPS. The SPS captures the spatio-temporal progression of the high-level event. For the example event depicted in Fig. 6 the SPS is “4i,3i,2i,1i,5i, 4t,3t,2t,1t,5t”.

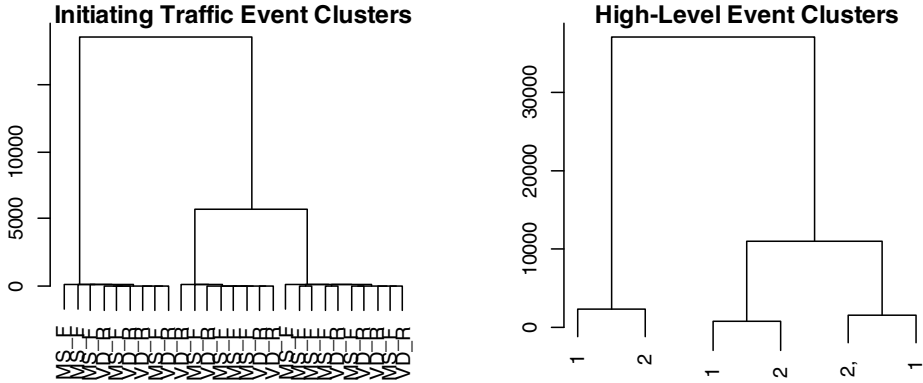


Fig. 7. Dendrogram of clustered I-events (MS_F- falling mean speed and VD_R-rising vehicle density) and 7b clustered I (1) and T(2) event temporal means which indicate candidate high-level events

5 Case Study: Storm Detection

To evaluate the approach with real data, we applied it to storm detection using meteorological time series data from ocean observing buoys in the Gulf of Maine (GoMOOS). The GOMOOS system collects data in a harsh marine environment so the sensor time series contain missing values due to bad weather and occasional service lags. A timeframe was chosen from 01-Oct-2004 22:00 to 04-Jul-2007 00:00 to minimize missing data periods. This timeframe included observed data for ten buoys. We used just a single marker parameter, barometric pressure, in this case because of the expected strong relationship of low pressure dynamics to storm formation. The barometric pressure time series were first smoothed using a low-pass filter and I-events (BP_fall) were extracted from the time series first derivative using a threshold of 4mbs per hour over a 6 hour duration. T-events (BP_rise) used a

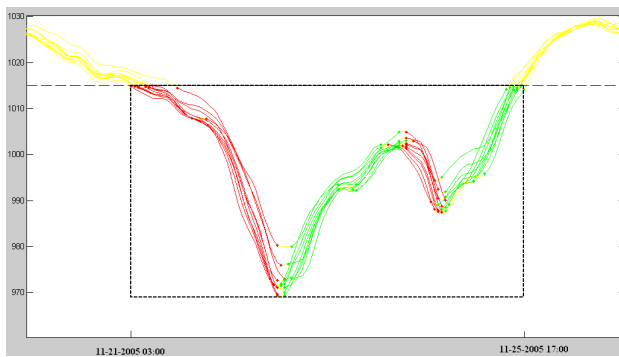


Fig. 8. Time series view of two pairs of I-event (red) and T-event (green) clusters for a candidate storm

threshold of 3 mbs per hour over a 3 hour duration. I-events and T-events were temporally clustered and the cluster means clustered to identify candidate storm events. Candidate storms were identified by at least one BP_fall primitive event cluster followed by a BP_rise primitive event cluster. Fig. 8 shows a time series view of a candidate storm with two cluster pairs of BP_Fall and BP_Rise primitive events.

A total of 113 candidate storms were identified within the study time window. To evaluate these results, we compared these derived candidate storms with storms reported in the National Climatic Data Center (NCDC) Storm Events database (<http://www4.ncdc.noaa.gov/cgi-win/wwcgi.dll?wwEvent~Storms>). This database includes information on storm start and end times, location and observations on the intensity and direction of low pressure movement. The NCDC database included 80 storms for this period. Seventy-one derived storms agreed with NCDC storms, while 39 had no match in the NCDC Storm Events database. NCDC identified nine storms which were not detected by the primitive event (PE) approach. Reasons for missing NCDC identified storms were in part due to missing sensor data periods (in a number of cases the pressure sensors failed in severe winter storms) and location differences. NCDC reported storms are based on land meteorological stations while the GoMOOS buoy locations are a distance off shore. Reasons for the extra derived storms are that the detected barometric pressure falls might correspond to low pressure systems that did not qualify as NCDC storm events and also the buoys locations could be detecting storms that tracked further out to sea and were not detected by land based stations.

An SPS for a candidate storm detected starting at 04-12-2007 17:00 and ending at 04-20-2007 15:00 was compared to a significant storm recorded by NOAA on April 15 2007. The storm description in the NCDC database stated: "An area of low pressure rapidly intensified while tracking from the southeastern states to the southern New England coast from April 15th to the 16th. A tight pressure gradient developed between the low and high pressure centered over eastern Canada which also blocked the northern movement of the low. The intense low slowly drifted east from the 16th through the 19th while high pressure remained across eastern Canada." The SPS for this candidate storm is represented in a graphic subset in Fig. 9 where red symbols indicated I-events and green indicate T-events and location IDs are the GOMOOS buoy locations named by letters. Graphic representation of the SPS by time slice shows falling barometric pressure events first detected by buoys A, B in the first time slice followed by detection at buoys C, E and F in the next time slice and eventually buoys A, B, C, E, F, I, L, M, N, see I-events as the storm tracks southwest to northeast. Later time slices of the SPS show retreat of the storm as the last terminating (rising barometric pressure) events were seen at the eastern most buoy locations (L, M and N).

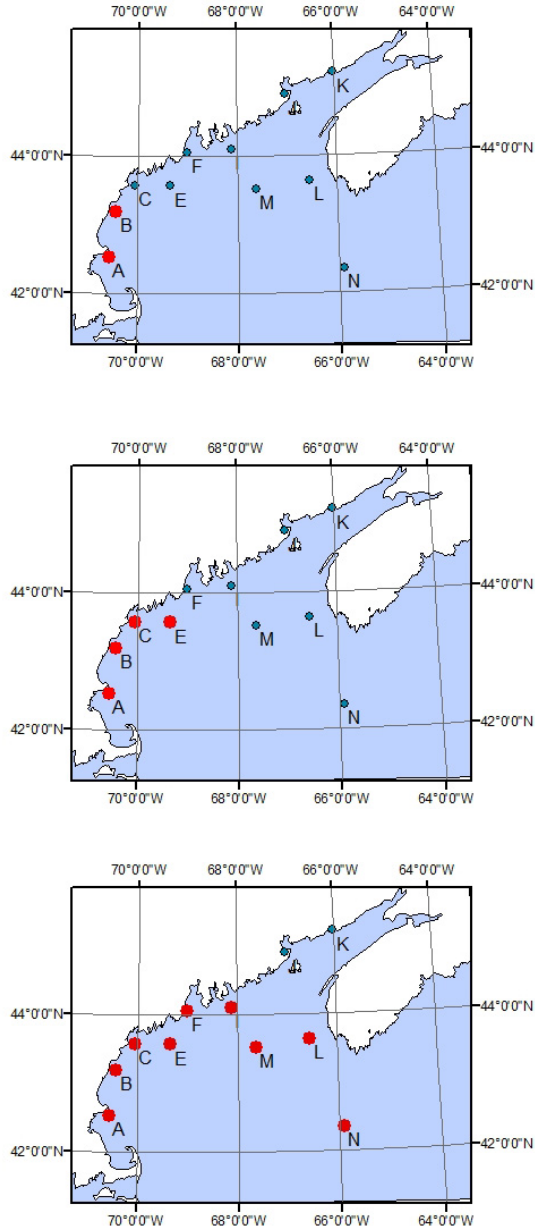


Fig. 9. Graphic representation of SPS for a candidate April 2007 storm

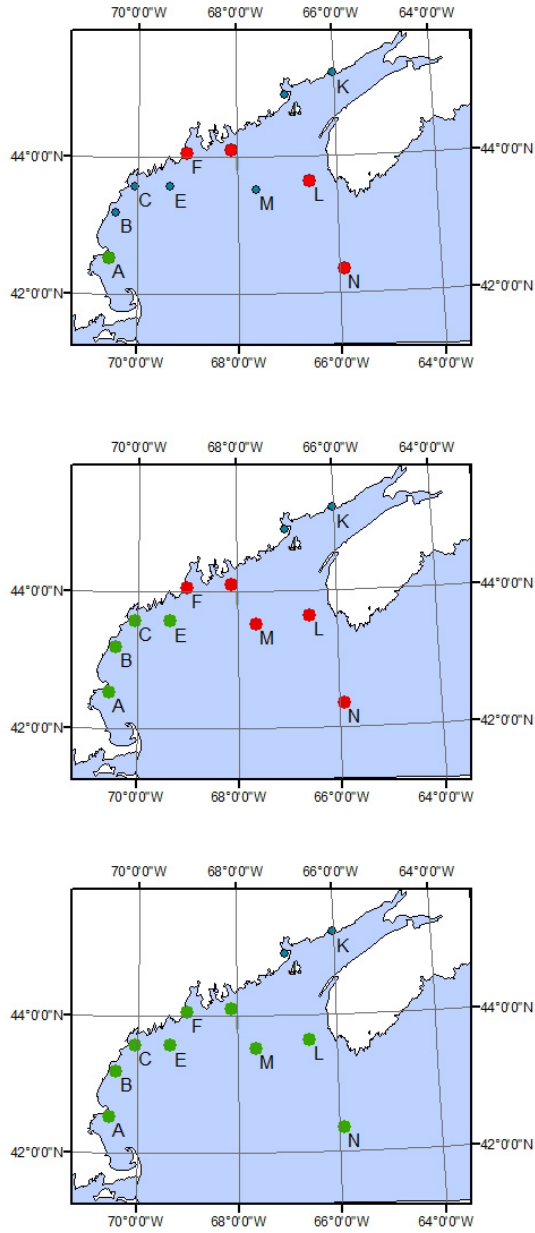


Fig. 9. (continued)

6 Summary and Future Work

This paper demonstrates an approach for high-level event detection using primitive events detected from time series distributed in space as building blocks. The approach applied to real geolocated time series data showed promise in identifying storm events and their spatial progression patterns over a sensed region. The approach also indicates the need for good correlation of a marker parameter with expected initiating conditions for a high level event. Future work will investigate more complex multivariate initiating conditions as well as more complex temporal initiating and terminating conditions (e.g temporal lags). Over a small region, clustering on the time stamps appeared sufficient, however a larger region might experience more than one high level event of the same type at similar times, in which case clustering should include spatial locations. In future work we plan to investigate fuzzy primitive event detection and the implications for high-level event detection. The case study applied the methodology to historic time series while in network, near real-time, primitive event and high-level event detection approaches would be of interest.

Acknowledgments. This material is based upon work supported by the National Science Foundation under Grant No. 0429644.

References

1. Fuentes, M., Reich, B., Lee, G.: Spatio-temporal meso-scale modeling of rainfall intensity using gage and radar data. *Annals of Applied Statistics* 2(4), 1148–1169 (2008)
2. Hussein, I., Spock, E., Pilz, J., Yu, H.-W.: Spatio-temporal interpolation of precipitation during monsoon periods in Pakistan. *Advances in Water Resources* 33(8), 880–886 (2010)
3. Jiang, J., Nittel, S., Worboys, M.: Qualitative change detection using sensor networks based on connectivity information. *GeoInformatica* 15(2), 305–328 (2011)
4. Neill, D.B.: Expectation based scan statistics for monitoring spatial time series data. *International Journal of Forecasting* 25, 498–517 (2009)
5. Guralnik, V., Srivastava, J.: Event detection from timeseries data. In: *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 33–42. ACM, New York (1999)
6. Alon, J., Sclaroff, S., Kollios, G., Pavlovic, V.: Discovering clusters in motion time-series data. In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. (1), pp. 375–381 (2003)
7. Allan, J., Papka, R., Lavrenko, V.: On-line new event detection and tracking. In: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information retrieval*, Melbourne, pp. 37–45. ACM, New York (1998)
8. Padmanabhan, B., Tuzhilin, A.: Pattern Discovery in Temporal Databases: A Temporal Logic Approach. In: *International Conference on Knowledge Discovery and Data Mining*, pp. 351–354 (1996)
9. Keogh, E., Chu, S., Hart, D., Pazzani, M.: Segmenting Time Series: A Survey and Novel Approach. In: *Data Mining in Time Series Databases*. World Scientific (2003)
10. Weigend, A.S., Gershenfeld, N.A.: Timeseries prediction: Forecasting the future and understanding the past. In: *Proceedings of the NATO Advanced Research Workshop on Comparative Time Series Analysis*. Addison-Wesley, Santa Fe (1994)

11. Höppner, F.: Discovery of Core Episodes from Sequences. In: Hand, D.J., Adams, N.M., Bolton, R.J. (eds.) *Pattern Detection and Discovery*. LNCS (LNAI), vol. 2447, pp. 199–213. Springer, Heidelberg (2002)
12. Morchen, F., Ultsch, A.: Discovering temporal knowledge in multivariate time series. In: Weihs, C., Gaul, W. (eds.) *Classification- the Ubiquitous Challenge*, pp. 272–279. Springer, Heidelberg (2005)
13. Abonyi, J., Feil, B., Nemeth, S., Avra, P.: *Principal Component Analysis based time series segmentation: a new sensor fusion algorithm (2004)* (preprint)
14. Batal, I., Sacchi, L., Bellazzi, R., Hauskrecht, M.: Multivariate time series classification with temporal abstractions. In: *Proceedings of the Twenty-Second International FLAIRS Conference (2009)*
15. Kuldorff, M.: A spatial scan statistic. *Communications in Statistics: Theory and Methods* 26(6), 1481–1496 (1997)
16. Velipasalar, S., Brown, L., Hampapur, A.: Specifying, Interpreting and Detecting High-level, Spatio-Temporal Composite Events in Single and Multi-Camera System. In: *Int'l Workshop on Semantic Learning Applications in Multimedia (SLAM), The Proceedings of IEEE CVPR Workshop, New York (2006)*
17. Spiegel, S., Gaebler, J., Lommatzsch, A., De Luca, E., Albayrak, S.: Pattern recognition and classification for Multivariate Time Series. In: *SenorKDD 2011, San Diego, CA, August 21 (2011)*
18. Shahar, Y.: A Framework for Knowledge-Based Temporal Abstraction. *Artificial Intelligence* 90, 79–133 (1997)
19. Abadi, D., Madden, S., Lindner, W.: REED: Robust, efficient filtering and event detection in sensor networks. In: *Proceedings of the 31st International Conference in Very Large Database (VLDB), Trondheim, Norway, pp. 769–780 (August 2005)*
20. Xue, W., Luo, Q., Chen, L., Liu, Y.: Contour map matching for event detection in sensor networks. In: *Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD), Chicago, IL, USA, pp. 145–156 (June 2006)*
21. Li, M., Liu, Y., Chen, L.: Non-threshold based event detection for 3d environment monitoring in sensor networks. In: *Proceedings of the 27th International Conference on Distributed Computing Systems, Toronto, Ontario, Canada, pp. 9–16 (June 2007)*
22. Wang, T.Y., Yu, C.T.: Collaborative event region detection in wireless sensor networks using markov random fields. In: *Proceedings of the 2nd International Symposium on Wireless Communication Systems (ISWCS), Siena, Italy, pp. 493–497 (September 2005)*
23. Wang, X.R., Lizier, J.T., Obst, O., Prokopenko, M., Wang, P.: Spatiotemporal Anomaly Detection in Gas Monitoring Sensor Networks. In: Verdone, R. (ed.) *EWSN 2008*. LNCS, vol. 4913, pp. 90–105. Springer, Heidelberg (2008)
24. Yin, J., Hu, D.H., Wang, Q.: Spatio-Temporal Event Detection Using Dynamic Conditional Random Fields. In: *IJCAI 2009 Proceedings of the 21st International Joint Conference on Artificial Intelligence, pp. 1321–1326 (2009)*
25. Galton, A., Mizoguchi, R.: The water falls but the waterfall does not fall: New perspectives on objects, processes and events. *Applied Ontology* 4(2), 71–107 (2009)
26. Allen, F.: Towards a general theory of action and time. *Artificial Intelligence* 23, 123–154 (1984)
27. Lin, F., Xie, K., Song, G., Wu, T.: A Novel Spatio-temporal Clustering Approach by Process Similarity. In: *Sixth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2009, pp. 150–154 (2009)*
28. Zahn, C.T.: Graph-theoretical methods for detecting and describing Gestalt clusters. *IEEE Transactions on Computers* C-20(1), 68–86 (1971)

Towards Vague Geographic Data Warehouses

Thiago Luís Lopes Siqueira^{1,2}, Cristina Dutra de Aguiar Ciferri³,
Valéria Cesário Times⁴, and Ricardo Rodrigues Ciferri²

¹ São Paulo Federal Institute of Education, Science and Technology at São Carlos, IFSP,
13.565-905, São Carlos, SP, Brazil

² Computer Science Department, Federal University of São Carlos, UFSCar,
13.565-905, São Carlos, SP, Brazil

³ Computer Science Department, University of São Paulo at São Carlos, USP
13.560-970, São Carlos, SP, Brazil

⁴ Informatics Center, Federal University of Pernambuco, UFPE,
50.670-901, Recife, PE, Brazil

prof.thiago@ifsp.edu.br, cdac@icmc.usp.br,
vct@cin.ufpe.br, ricardo@dc.ufscar.br

Abstract. Currently, geographic data warehouses provide a means of carrying out spatial analysis together with agile and flexible multidimensional analytical queries over huge volumes of data. However, they do not enable the representation and neither the analysis over real world phenomena that have uncertain locations or vague boundaries, which are denoted by vague spatial objects. In this paper, we introduce the vague geographic data warehouse (vGDW) and its spatially-enabled components at the logical level: attributes, measures, dimensions, hierarchies and queries. We base the vGDW on exact models to represent vague spatial objects. In addition, we combine the fuzzy model with the exact model in relational vGDW to improve the expressiveness of the queries. Finally, a case study is presented to validate our contributions.

Keywords: geographic data warehouse, vagueness, logical modeling.

1 Introduction

Although geographic data warehouses (GDWs) provide a means of carrying out spatial analysis together with agile and flexible multidimensional analytical queries over huge volumes of data [1][2][3], little attention has been devoted to provide support for vague spatial objects in GDW. In addition, existing data models that represent vague spatial objects focus on spatiality and are not aimed to couple with data warehouses (DWs) [4][5][6][7]. As a result, current GDWs do not enable the representation and neither the analysis over real world phenomena that have uncertain locations or vague boundaries. Such phenomena are represented by vague spatial objects.

In this paper, we introduce the vague geographic data warehouse (vGDW). Our baseline is the star schema with fact and dimension tables implemented under the relational model (analogous with [8]). We reuse exact models to represent vague spatial

objects [7][9][10]. Furthermore, in this paper, we aim to: (i) define the components of the vGDW and state relevant issues of its logical design; (ii) demonstrate how to reuse DW and GDW concepts to design vGDWs; (iii) demonstrate how to reuse exact models to provide support for vague spatial objects in vGDWs; (iv) carry out the essential adaptations and extensions to enable the design and the implantation of the vGDW; (v) separately manipulate the certain and the vague components of vague spatial objects in the vGDW; and (vi) combine distinct approaches that model and represent vague spatial objects in vGDW, i.e. combine the fuzzy model with the exact model in relational vGDW to improve the expressiveness of the queries.

The remainder of this paper is organized as follows. Section 2 summarizes the theoretical foundation. Section 3 introduces the vGDW. Section 4 presents a case study. Section 5 describes the customization of the vGDW design to achieve the cited goals (iv) and (v). Section 6 addresses related work. Section 7 concludes the paper.

2 Theoretical Foundation

2.1 Vague Spatial Data

In contrast to crisp spatial objects, vague spatial objects are used to represent real world phenomena that do not have exact locations or well-defined boundaries [4][5][6][7]. Vague spatial objects can be modeled according to four main approaches: (i) probabilistic models [4][6][11]; (ii) fuzzy models [4][5][12]; (iii) rough sets models [13][14]; and (iv) exact models [7][9][10]. In this paper, we focus on exact models because they reuse and adapt the research legacy for crisp spatial data in order to manipulate vague spatial data. This legacy is implemented in the spatial extensions of database management systems (DBMS) [15], and therefore we can apply and adapt it to maintain and manipulate vGDWs. Recently, two exact models were proposed and gained our attention to be addressed in this paper: QMM [9][10] and VASA [7].

The Qualitative Min-Max Model (QMM) defines a vague spatial object as a pair of crisp complex spatial objects, namely the minimum extent and the maximal extent. The former is the determinate element of the vague spatial object. On the contrary, the latter is the vague element of the vague spatial object. If a point p belongs to the minimum extent, then it also belongs to the spatial object. However, if p belongs to the maximal extent, then *maybe* p belongs to the spatial object. Finally, if p does not belong to the minimum extent and neither to the maximal extent, then p does not belong to the spatial object. Adverbial qualifiers describe a vague spatial object, according to the existence of vagueness in its boundary or interior. For instance, a fairly vague line has crisp boundaries and a complete vague interior. A vague point is represented by a crisp region. A vague line has the interior or the boundary represented by crisp regions. Finally, a vague region is composed of a pair of crisp regions, such that the maximal extent can be equal to, contain, or cover the minimal extent.

The Vague Spatial Algebra (VASA) defines a vague spatial object as a pair of crisp complex spatial objects: the kernel and the conjecture. The kernel is similar to the QMM's minimum extent, while the conjecture is similar to the QMM's maximum extent. However, the following two main restrictions differs VASA from QMM: the interior of the kernel is disjoint from the interior of the conjecture; also, the kernel and the conjecture necessarily belong to the same data type. Both VASA and QMM definitions are reused, adapted and extended in Section 3.1.

2.2 Geographic Data Warehouse

A geographic data warehouse (GDW) provides a means of carrying out spatial analysis together with agile and flexible multidimensional analytical queries over huge volumes of data [1][2]. A GDW implemented in a relational database inherits several components of conventional data warehouses, such as fact and dimension tables, numeric measures and hierarchies of attributes that aggregate these measures according to distinct granularity levels [3]. Fact tables store numeric measures that indicate the scores of business activities, while dimension tables have attributes that describe and group the values of these measures. Additionally, the GDW has spatial attributes that store crisp spatial objects implemented as vector geometries to compose spatial dimension tables, spatial measures and spatial hierarchies [1].

A hierarchy of attributes is a predefined association among higher and lower granularity attributes that is determined by a spatial relationship [1]. For example, $city \preceq address$ with the cardinality of 1:N and the containment relationship, and $highway \preceq state$ with the cardinality of M:N and the intersection relationship. According to [16], $Q_1 \preceq Q_2$ if, and only if it is possible to answer Q_1 using just the results of Q_2 , and $Q_1 \neq Q_2$.

The well-known star and snowflake schemas may be adequately adapted to support the inclusion of spatial attributes, which introduce new storage costs and might impair query processing performance [8]. Several previous work have proven the infeasibility of storing geometries in the fact table, because: (i) geometries call for a varying storage space according to their shapes, and then require a selective materialization [17]; (ii) the spatial data redundancy impairs the query processing performance in GDW and must be avoided [18]; and (iii) considering a table that holds conventional and spatial attributes, as the number of vertices of spatial objects is increased, also the response time of queries increase [19]. An alternative is to apply the OLAP operator *pull* to spatial measures. This operator converts a measure into a dimension [20]. In addition to the *pull* operator, identifiers as surrogate keys are created for the spatial objects, and spatial data redundancy is avoided [18]. Finally, spatial dimension tables should be designed, preferentially, with only a pair of attributes: the primary key attribute and the spatial attribute [21].

In GDWs, spatial online analytical processing (SOLAP) queries hold spatial predicates, e.g.: *roll-up*, *drill-down*, *slice-and-dice* and *drill-across* [3][22]. In addition to range queries, other types of queries that can be submitted to GDW to assess spatial attributes are: spatial join [23] and nearest neighbor queries [24]. In Section 3, we reuse and adapt the concepts described in this section to apply on the vGDW.

3 Vague Geographic Data Warehouses

In this section, we define a *vague geographic data warehouse* in Definition 1. Furthermore, Section 3.1 describes attributes such as the vague spatial attribute. In Section 0, spatial dimensions and spatial measures are detailed. Hierarchies are introduced in Section 3.3. Finally, queries are stated in Section 3.4.

Definition 1. A *vague geographic data warehouse* (vGDW) is a data warehouse that has at least one vague spatial attribute, which is held by a fact or a dimension table.

3.1 Attributes

In vGDWs, attributes are classified as *conventional attributes* or *spatial attributes*. The former have numeric or alphanumeric domains, while the latter comprise vector geometries to locate and describe the shape of the real world phenomenon. Regarding *spatial attributes*, they are sub classified as *crisp spatial attributes* and *vague spatial attributes*. The domain of a *crisp spatial attribute* is composed of spatial objects that represent real world phenomena that have exact locations and well-known boundaries. On the other hand, a *vague spatial attribute* holds *vague spatial objects* in its domain. Our definition for vague spatial objects inherits, combines and extends the characteristics of vague spatial objects defined by VASA and QMM, as follows.

In vGDWs, a *vague spatial object* consists of a vector representation composed of a pair of crisp spatial objects, namely the *core* and the *dubiety*. Both the core and the dubiety may assume simple or complex shapes. Their interiors are disjoint. This constraint avoids the redundant storage of points. Also, the core is not necessarily located in the center of the spatial object, despite its name. Furthermore, vague spatial objects can also follow these properties:

- If the core is empty, then the object is completely vague;
- If the dubiety is empty, then the object is crisp;
- The core and the dubiety are not necessarily of the same data types; and
- The core and the dubiety may be disjoint.

Moreover, some of the objects in the domain of a vague spatial attribute may be crisp. This is true when some of the instances of a given phenomenon has unknown locations or vague boundaries, while other instances are precisely located as well as have well-defined boundaries. For example, suppose that the habitats of some animals are being mapped. The habitat for a given animal can be mapped as a crisp region (e.g. an island where monkeys live), while for other animals the habitat may be mapped as a region with vague boundaries (e.g. an area where ants live). Therefore, we also classify attributes according to the percentage of vague spatial objects that exist in its domain, as shown in Table 1.

Table 1. The classification of vague spatial objects

Percentage of vague spatial objects	Classification
0%	Crisp spatial attribute
(0% – 100%)	Partially vague spatial attribute
100%	Completely vague spatial attribute

With the classification shown in Table 1, the designer and the administrator of the vGDW are able to model the vGDW aiming to improve the query processing performance, considering the spatial objects that are held by a given attribute. Note that, as new objects are added to the domain of a given attribute, its classification may change.

3.2 Measures and Dimensions

The spatial attributes described in Section 3.1 can be applied to both fact and dimension tables of a vGDW. If the spatial attribute is held by the fact table, then it is considered a *spatial measure*. A spatial measure determines a location to geographically describe each row of the fact table. However, as mentioned in Section 2.2, storing geometries in the fact table is infeasible. Therefore, spatial measures must be converted into a spatial dimension. In addition, identifiers as surrogate keys are created for the involved spatial objects, and spatial data redundancy must be avoided.

Conversely, if a spatial attribute is held by a dimension table, then it is considered a *spatial dimension*. Spatial dimensions are sub classified in *factual spatial dimension table* and *dimensional spatial dimension table*. When the spatial dimension table is referenced by the fact table through a foreign key, it is considered *factual*, since it describes the fact together with the other dimension tables that the fact table references. On the other hand, if the spatial dimension table is referenced by another dimension table, then it is considered *dimensional*, since it provides a geographic description for another dimension.

3.3 Hierarchies

Attributes in the same or in different dimension tables may be related through hierarchies denoted as $A_1 \leq \dots \leq A_n$ and assume, for each pair of attributes A_i and A_{i+1} , that $A_i \leq A_{i+1}$ (with $0 < i \leq n$). In the vGDW context, A_i and A_{i+1} can be classified as shown in Table 2 and a hierarchy associating n attributes can hold attributes of distinct classifications. Therefore, according to the classifications shown in Table 2, the hierarchies can be categorized as shown in Table 3.

Table 2. Classifications for the attributes of a hierarchy $A_i \leq A_{i+1}$

A_i	A_{i+1}
Conventional attribute	Conventional attribute
Conventional attribute	Crisp spatial attribute
Conventional attribute	Vague spatial attribute
Crisp spatial attribute	Conventional attribute
Crisp spatial attribute	Crisp spatial attribute
Crisp spatial attribute	Vague spatial attribute
Vague spatial attribute	Conventional attribute
Vague spatial attribute	Crisp spatial attribute
Vague spatial attribute	Vague spatial attribute

Table 3. How hierarchies of attributes can be categorized in vGDWs

Classifications of the attributes in the hierarchy	Hierarchy category
Only conventional attributes	Non-spatial hierarchy
Only crisp spatial attributes	Crisp spatial hierarchy
Only vague spatial attributes	Vague spatial hierarchy
Conventional attributes and crisp spatial attributes	Partially crisp spatial hierarchy
Conventional attributes and vague spatial attributes	Partially vague spatial hierarchy
Crisp spatial attributes and vague spatial attributes	Completely spatial hierarchy
Conventional attributes, crisp spatial attributes, vague spatial attributes	Hybrid hierarchy

3.4 Queries

Queries that are submitted to vGDW require joining fact and dimension tables, performing filters to retrieve specific values, and finally group and sort the results to adequately present them. Regarding filters, they may concern conventional or spatial attributes. When dealing with conventional attributes, both exact match and range queries can be issued. On the other hand, when dealing with spatial attributes, well-known range queries predominate, e.g. *intersection range query*, *containment range query* and *enclosure range query*. These range queries relate a spatial attribute to a given query window whose shape does not belong to the domain of any of the existing spatial attributes, i.e. an *ad hoc* spatial query window. Exact match queries are not common because they could be implemented as a conventional predicates.

Specifically for vague spatial attributes, we have defined the *vague range query* (VRQ) [19]. It uses a vague region to assess a containment range query considering its core and an intersection range query considering its dubiety. As a result, two predicates are required: one is the more restrictive and issued on the certain component of the query window (i.e. the containment range query issued on the core) and the other is less restrictive and issued on the vague component of the query window (i.e. the intersection range query issued on the dubiety).

Regarding SOLAP operations, a *drill-down* operation occurs with data aggregation firstly in an upper level and later in a lower level. On the other hand, a *roll-up* operation occurs if data aggregation is done inversely. These both operations must take into account the classification of the attributes involved. For instance, if a query is submitted and evaluates a spatial predicate on the crisp spatial attribute A_i , this spatial predicate may not be available for analysis when drilling-down to the conventional attribute A_{i+1} . Suppose that A_i refers to continents and A_{i+1} refers to countries. Therefore, the *within* spatial predicate referring to the map of North America would be replaced by a conventional predicate such as *IN* {'Canada', 'Mexico', 'USA', ...}, when drilling-down to countries. A similar treatment would be given if the pair of attributes A_i and A_{i+1} , involved in the *drill-down/roll-up* operation, were any of those listed in Table 2. The same treatment would also be given for any pair of attributes A_i and A_j , such that $j > i + 1$ and $j \leq n$.

In addition, filters applied to conventional or spatial attributes enable the *slice-and-dice* operation. To execute a *drill-across* operation, spatial predicates are applied to the comparison between the values of two distinct measures that belong to different fact tables. These fact tables essentially reference one or more dimension tables simultaneously. Finally, spatial joins and nearest neighbor queries are also supported.

4 Case Study: Pest Control

The following application describes a vGDW that was built aiming at monitoring the pest control over plantations and crop parcels. Fig. 1 shows the spatial objects that represent a rural area, the crop parcels, the plantations and one irrigation channel. Fig. 1a highlights one crop parcel that is detailed in Fig. 1b.

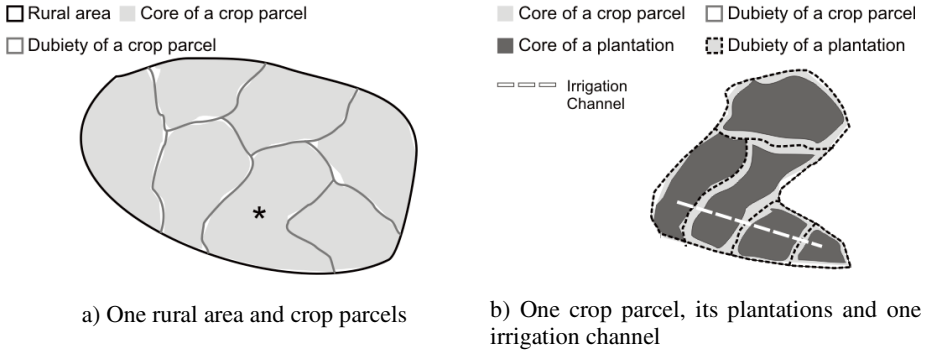


Fig. 1. Spatial objects representing the pest control application

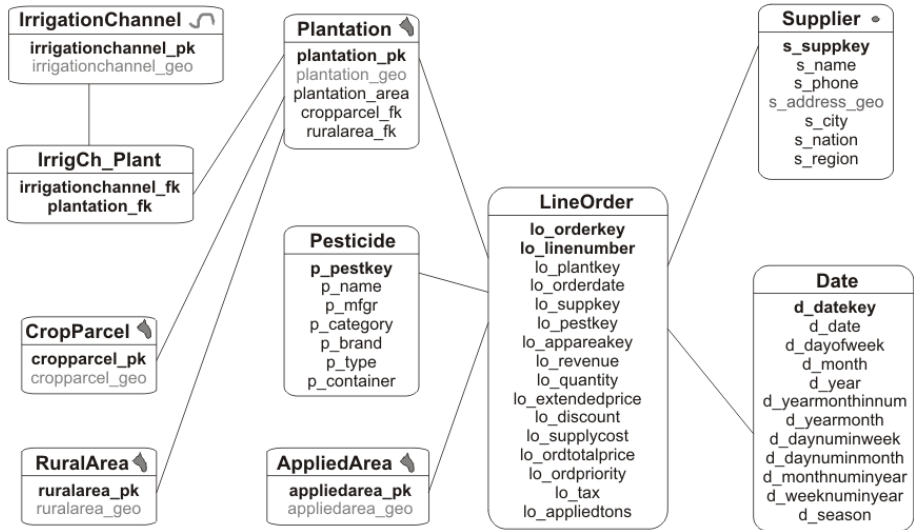


Fig. 2. A vGDW in the context of pest control

The vGDW was built as shown in Fig. 2, reusing the pictograms provided by the MultiDimER Model to indicate the spatial data type used in each table [1]. Although MultiDimER is suitable for conceptual modeling, we are interested in the logical model that is an extension of the conventional star schema. The fact table *LineOrder*

references the dimension tables through foreign keys. It also holds measures such as *lo_appliedtons* that denotes the amount of pesticides in tons that was applied. The conventional dimension tables are *Date* and *Pesticide*. In addition, there are six spatial dimension tables: *Supplier*, *Plantation*, *IrrigationChannel*, *CropParcel*, *RuralArea* and *AppliedArea*. In Section 4.1, the attributes and objects maintained by this vGDW are described. In Section 4.2, spatial dimensions and spatial measures are detailed. Hierarchies are described in Section 4.3. Furthermore, queries are stated in Section 4.4. We encourage readers to check the SQL implementation of this vGDW that is under <http://gbd.dc.ufscar.br/pestcontrolygdw>.

4.1 Attributes

In Table 4, the spatial attributes of the vGDW shown in Fig. 2 are detailed in terms of: the classification given in Section 4.1, the spatial data types and the DBMS data types. We utilized PostGIS (<http://postgis.refractory.net>) because it is a well-known and widely used spatial extension for databases. The remaining attributes that were not mentioned in Table 4 are conventional. As a result, this vGDW maintains crisp and vague spatial objects to geographically describe the phenomena, as well as conventional attributes to characterize them. If, for instance, one crisp region was added to represent a plantation, then *plantation_geo* would become a partially vague spatial attribute, according to Table 1.

Table 4. The classification for the attributes of the vGDW shown in Fig. 2

Attribute	Classification	Data type	DBMS data type
s_address_geo	crisp spatial attribute	simple point	POINT
plantation_geo	completely vague spatial attribute	complex polygon	MULTIPOLYGON
irrigation_channel_geo	crisp spatial attribute	simple line	LINestring
cropparcel_geo	completely vague spatial attribute	complex polygon	MULTIPOLYGON
ruralarea_geo	crisp spatial attribute	simple polygon	POLYGON
appliedarea_geo	completely vague spatial attribute	complex polygon	MULTIPOLYGON

4.2 Measures and dimensions

The vGDW shown in Fig. 2 contains three factual spatial dimension tables: *Supplier*, *Plantation* and *AppliedArea*. Therefore, each tuple of the fact table is associated to a supplier address, a plantation region and an area where a pesticide was applied. The table *AppliedArea* is a spatial measure that was *pulled* into a dimension table to preserve the performance of the query processing and the storage requirement.

Concerning dimensional spatial dimension tables, i.e., *IrrigationChannel*, *CropParcel* and *RuralArea*, they provide extended geographic descriptions for plantations. Also, the table *RuralArea* is referenced by the table *Plantation* instead of being referenced by the table *CropParcel*, aiming to avoid a schema similar to a snowflake schema, which would introduce more joins between these tables.

4.3 Hierarchies

The GDW depicted in Fig. 2 also holds some hierarchies that associate attributes of different data types. Also, these hierarchies have particular cardinalities and refer to different topological relationships. Table 5 lists some of the existing hierarchies. The existence of attributes with different data types demands the correct treatment to perform *roll-up/drill-down* operations, as described in Section 3.3. For example, when *drilling-down* from *s_address_geo* to *s_suppkey*, it is necessary to dispense the spatial predicate and replace it by a proper conventional predicate.

Table 5. Some hierarchies of attributes found in the vGDW depicted in Fig. 2

Hierarchy	Cardinality	Hierarchy category	Topological relationship
$p_mfgr \preceq p_category \preceq p_brand \preceq p_pestkey$	1:N	Non-spatial hierarchy	-
$s_address_geo \preceq s_suppkey$	1:1	Partially crisp spatial hierarchy	Containment
$ruralarea_geo \preceq cropparcel_geo \preceq plantation_geo$	1:N	Completely Spatial hierarchy	Containment
$irrigationchannel_geo \preceq plantation_geo$	M:N	Completely Spatial hierarchy	Intersection

4.4 Queries

Fig. 3 shows a template for the queries applied to the vGDW of Fig. 2. The gaps 1 and 2 are a list of tables and a list of joins/filters criteria, respectively. These gaps are described in Table 6, which details three different queries. For instance, queries Q1 and Q2 are intersection range queries that consider an ad hoc spatial query window q . In addition, since $ruralarea_geo \preceq cropparcel_geo \preceq plantation_geo$, executing Q1 and Q2 consecutively determines a *drill-down* operation, while the inverse execution determines a *roll-up* operation. Query Q3 interestingly involves the spatial measure. Finally, another query of this pest control application is given in Fig. 4: Q4 is a VRQ that uses $q1$ and $q2$ as spatial query windows that are quadratic and concentric. Also, $area(q1) < area(q2)$ and $q1$ is within $q2$. The labels ‘More relevant’ and ‘Less relevant’ in the select clause rank the results in more and less relevant, respectively.

```

SELECT d_year, s_nation,
       SUM(lo_revenue - lo_supplycost) AS profit, SUM(lo_appliedtons) AS pesticide_tons
FROM Date, Supplier, Pesticide, GAP 1, LineOrder
WHERE lo_custkey = c_custkey AND lo_suppkey = s_suppkey
  AND lo_pestkey = p_pestkey AND lo_orderdate = d_datekey
  AND GAP 2 AND s_region = 'AMERICA'
  AND (p_mfgr = 'MFGR#1' OR p_mfgr = 'MFGR#2')
GROUP BY d_year, s_nation
ORDER BY d_year, s_nation;

```

Fig. 3. A template for the queries issued over the vGDW shown in Fig. 2

Table 6. Filling the gaps of the template shown in Fig. 3, to produce the queries

Query	Gap 1	Gap 2
Q1	Plantation , RuralArea	lo_plantkey = plantation_pk AND ruralarea_fk = ruralarea_pk AND INTERSECTS (q, ruralarea_geo)
Q2	Plantation, CropParcel	lo_plantkey = plantation_pk AND cropparcel_fk = cropparcel_pk AND INTERSECTS (q, cropparcel_geo)
Q3	Plantation, IrrigCh_Plant, IrrigationChannel	lo_plantkey = plantation_pk AND plantantion_pk = plantation_fk AND irrigationchannel_fk = irrigationchannel_pk AND lo_appareakey = appliedarea_pk AND ST_Distance(irrigationchannel_geo, lo_appliedarea_geo) < 50

```

SELECT SUM (lo_revenue), d_year, p_brand, 'More relevant'
FROM LineOrder, Date, Pesticide, Plantation
WHERE lo_orderdate = d_datekey AND lo_pestkey = p_pestkey
AND lo_plantkey = plantation_pk AND p_brand = 'MFGR#2239'
AND WITHIN(plantation_geo, q1)
GROUP BY d_year, p_brand
ORDER BY d_year, p_brand
UNION
SELECT SUM (lo_revenue), d_year, p_brand, 'Less relevant'
FROM LineOrder, Date, Pesticide, Plantation
WHERE lo_orderdate = d_datekey AND lo_pestkey = p_pestkey
AND lo_plantkey = plantation_pk AND p_brand = 'MFGR#2239'
AND INTERSECTS(plantation_geo, q2) AND NOT WITHIN(plantation_geo, q1)
GROUP BY d_year, p_brand
ORDER BY d_year, p_brand;

```

Fig. 4. A VRQ over the vGDW shown in Fig. 2

5 Customizing the Design of Vague Geographic Data Warehouses

In this section, we discuss two improvements over the design of vGDW. In Section 5.1, we describe the separate manipulation of core and dubiety to allow individual spatial predicates over them. In Section 5.2, we include fuzziness into a relational vGDW whose vague spatial objects were implemented using an exact model.

5.1 Manipulating Core and Dubiety Separately

The vGDW design discussed in Section 3 and exemplified in Section 4 considered vague spatial objects as a whole. As a result, one single spatial attribute stored both

the core and the dubiety of an object. Then, complex geometric data types were used to represent such attributes (Table 4). In addition, spatial predicates referred to the whole object when evaluating topological relationships. However, if core and dubiety need to be analyzed separately, the practices previously discussed are not adequate to provide such analysis. As a result, we argue that the core and the dubiety must have their own spatial attributes in the vGDW.

In order to provide this feature, the vague spatial dimension table is replaced by: one table that maintains the primary key and the conventional attributes of the former table, still called *vague spatial dimension table*; one table that stores the core into a crisp spatial attribute, called *core spatial dimension table*; and one table that stores the dubiety into a crisp spatial attribute, called *dubiety spatial dimension table*. The new tables reflect the 1:N associations among an object and its core parts, and among an object and its dubiety parts. Consequently, these tables are designed similarly to the mapping of a multivalued attribute. Finally, the core and dubiety new tables reference the primary key of the former vague spatial dimension. These transformations enable spatial predicates to evaluate the core or dubiety of an object separately, i.e. evaluate individually the certain component or the vague component of the vague spatial objects.

For instance, suppose that cores and dubieties of the areas where pesticides were applied, in Fig. 2, need to be analyzed separately. Fig. 5 exemplifies one applied area and its three dubiety parts (the values in brackets are detailed in Section 5.2). The former table *AppliedArea* is replaced by the following tables shown in Fig. 6: *AppliedArea*, *AppliedAreaCore* and *AppliedAreaDubiety*. The tables with suffix *Core* and *Dubiety* are the core spatial dimension table and the dubiety spatial dimension table, respectively. The underlined attributes compose the primary keys for these tables. The attributes with suffix *_fk* have foreign keys to *appliedarea_pk*, while those attributes with suffix *_geo* are crisp spatial attributes. As a result, the table *AppliedArea* does not store spatial objects anymore. Furthermore, each tuple of *AppliedAreaCore* and *AppliedAreaDubiety* refer to single geometries of simple spatial data type (e.g. POLYGON). For instance, there is one tuple for each one of the three parts that compose the dubiety of the applied area shown in Fig. 5. Moreover, these transformations enable spatial predicates to evaluate the vague boundary of the applied areas, such as *INTERSECTS* (q , *appliedaread_geo*), where q is an ad hoc spatial query window as shown in Fig. 5.

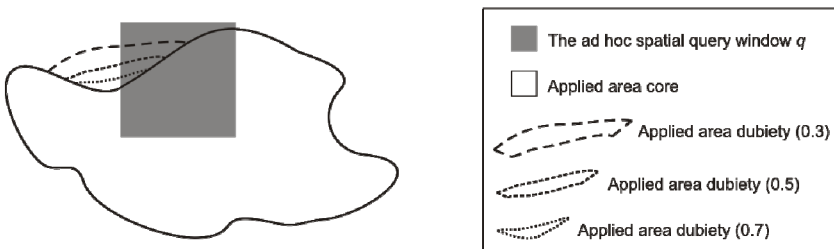






Fig. 5. An intersection range query involving dubiety parts

appliedarea_pk
1

core_id	appliedarea_fk	core_geo
1	1	

dublety_id	appliedarea_fk	dublety_geo
1	1	
2	1	
3	1	




dublety_id	appliedarea_fk	duclety_geo	dublety_fuzzy
1	1		0.7
2	1		0.5
3	1		0.3

Fig. 6. How the core and the dubiety parts of one pesticide applied area are stored separately

5.2 Providing Support to the Fuzzy Model

In contrast to the previous sections that addressed the design of vGDW using exact models, this section describes how to combine the fuzzy model with the exact model and relational vGDW in order to improve the expressiveness. As a result, reasoning about vague spatial objects does not lead only to a 3-valued logic (true, false or maybe) according to exact models, but now considers partial truths with membership degrees of the fuzzy model. The combination of these models is another relevant contribution of this paper. Such combination depends on the adaptations discussed in Section 5.1, because membership degrees apply exclusively to dubieties, not to cores.

In order to support the fuzzy model, one numeric attribute to denote the fuzzy membership degree value is included in the table that contains the dubiety spatial attribute. This attribute is called *fuzzy attribute*. The inclusion of such attributes associates each part of the dubiety to a membership value. Furthermore, it enables the evaluation of exact match or range queries over these values.

For instance, consider that the vGDW depicted in Fig. 2 was already modified as proposed in Section 5.1. Then, suppose that fuzzy membership degrees need to be associated to the vague boundaries of the pesticide applied areas. Consider Fig. 5 and the values of the membership degrees of each dubiety part in brackets. Then, the table *AppliedAreaDublety* must be replaced by the table *AppliedAreaDubletyFuzzy*. Both tables are shown in Fig. 6. As a result, it is feasible to evaluate predicates such as *INTERSECTS* (q , *appliedaread_geo*) *AND* *dublety_fuzzy* > 0.5, for example. In this particular case shown in Fig. 5, only the dubieties that intersect the ad hoc spatial query window q and that have a membership degree greater than 50% would be retrieved.

6 Related Work

Currently, existing conceptual, logical and physical designs for geographic data warehouses prioritize the support only for crisp spatial objects [1][2][3][17][25]. As a result, multidimensional analysis has not been coupled with spatial analysis of real world phenomena that are characterized by having uncertain locations or vague boundaries. Nevertheless, our work addresses this issue specifically at the logical design, and introduces the vGDW and its spatially-enabled components: attributes, measures, dimensions, hierarchies and queries.

Furthermore, existing data models to represent vague spatial objects focus on spatiality and were not aimed to couple with data warehouses [4][5][6][7][9][10][11][12][13][14]. However, in this paper our focus is to reuse, adapt and extend the existing exact models [7][9][10] to enable the vGDW. We argue that, as exact models inherit and adapt the research legacy involving crisp spatial objects, they are more suited to be applied to relational databases and to be implemented under the available DBMSs' spatial extensions. As a result, we introduce the core and the dubiety here, which are more generic components for spatial data objects than those of [7][9][10] and which were implemented using the DBMS. Moreover, we introduce the combination of the fuzzy model with the exact model, to improve the expressiveness of the vGDW queries.

The representation of field data in data warehouses had been formalized in [2][26]. However, they addressed conceptual models and did not validate the models in the context of relational databases. Conversely, this paper focuses on the logical design and on the use of vector data, which are directly applicable to existing DBMSs' spatial extensions. In addition, we present a case study to validate our definitions for the vGDW under the relational model.

Finally, although we had assessed the performance of vague spatial objects maintained by a geographic data warehouse once [19], we did not focus on the design of vGDWs and neither on the reuse and extension of existing exact models.

7 Conclusions and Future Work

In this paper, we have introduced the vague geographic data warehouse (vGDW) and its spatially-enabled components at the logical level: attributes, measures, dimensions, hierarchies and queries. We have demonstrated how to reuse, adapt and extend the research legacies from exact models, DW and GDW in order to design relational vGDWs. Furthermore, we have proposed the separate manipulation of the certain and the vague components of vague spatial objects in the vGDW. As another contribution, we have combined fuzzy models with exact models in relational vGDW, to improve the expressiveness of the queries.

As future work, we intend to design specific schemas to maintain the types of attributes and hierarchies that we defined, and then evaluate the query processing performance over these schemas. Also, we intend to define and assess the performance of spatially-enabled *drill-down*, *roll-up*, *slice-and-dice* and *drill-across* in vGDW.

References

1. Malinowski, E., Zimányi, E.: *Advanced Data Warehouse Design: From Conventional to Spatial and Temporal Applications*. Springer (2008)
2. Bimonte, S., Tchounikine, A., Miquel, M., Pinet, F.: *When Spatial Analysis Meets OLAP: Multidimensional Model and Operators*. In: Taniar, D., Iwan, L. (eds.) *Exploring Advances in Interdisciplinary Data Mining and Analytics*, pp. 249–277. IGI (2011)
3. Siqueira, T.L.L., Ciferri, C.D.A., Times, V.C., Ciferri, R.R.: *The SB-index and the HSB-Index: efficient indices for spatial data warehouses*. *Geoinformatica* 16(1), 165–205 (2011)

4. Burrough, P.A., Frank, A.U. (eds.): *Geographic Objects with Indeterminate Boundaries*. GISDATA, vol. 2. Taylor & Francis (1996)
5. Schneider, M.: *Fuzzy Spatial Data Types for Spatial Uncertainty Management in Databases*. In: *Handbook of Research on Fuzzy Information Processing in Databases*, pp. 490–515. IGI (2008)
6. Yuen, S., Tao, Y., Xiao, X., Pei, J.: *Superseding Nearest Neighbor Search on Uncertain Spatial Databases*. *TKDE* 22(7), 1041–1055 (2010)
7. Pauly, A., Schneider, M.: *VASA: An algebra for vague spatial data in databases*. *Inf. Syst.* 35(1), 111–138 (2010)
8. Kimball, R., Ross, M.: *The Data Warehouse Toolkit*, 2nd edn. Wiley (2002)
9. Bejaoui, L., Pinet, F., Schneider, M., Bédard, Y.: *OCL for formal modelling of topological constraints involving regions with broad boundaries*. *GeoInformatica* 14(3), 353–378 (2010)
10. Bejaoui, L., Pinet, F., Bédard, Y., Schneider, M.: *Qualified topological relations between spatial objects with possible vague shape*. *IJGIS* 23(7), 877–921 (2009)
11. Cheng, R., Kalashnikov, D.V., Prabhakar, S.: *Evaluating Probabilistic Queries over Imprecise Data*. In: *SIGMOD Conference*, pp. 551–562 (2003)
12. Dilo, A., de By, R.A., Stein, A.: *A System of Types and Operators for Handling Vague Spatial Objects*. *IJGIS* 21(4), 397–426 (2007)
13. Bittner, T., Stell, J.G.: *Vagueness and Rough Location*. *Geoinformatica* 6(2), 99–121 (2002)
14. Worboys, M.: *Computation with imprecise geospatial data*. *Computers, Environmental and Urban Systems* 22(2), 85–106 (1998)
15. Egenhofer, M.J., Franzosa, R.D.: *Point-set Topological Spatial Relations*. *IJGIS* 5(2), 161–174 (1991)
16. Harinarayan, V., Rajaraman, A., Ullman, J.D.: *Implementing Data Cubes Efficiently*. *ACM SIGMOD Record* 25(2), 205–216 (1996)
17. Stefanovic, N., Han, J., Koperski, K.: *Object-Based Selective Materialization for Efficient Implementation of Spatial Data Cubes*. *TKDE* 12(6), 938–958 (2000)
18. Siqueira, T.L.L., Ciferri, R.R., Times, V.C., Ciferri, C.D.A.: *The Impact of Spatial Data Redundancy on SOLAP Query Performance*. *JBCS* 15(2), 19–34 (2009)
19. Siqueira, T.L.L., Mateus, R.C., Ciferri, R.R., Times, V.C., Ciferri, C.D.A.: *Querying Vague Spatial Information in Geographic Data Warehouses*. In: *AGILE Conference*, pp. 379–397 (2011)
20. Pourabbas, E., Rafanelli, M.: *Characterization of Hierarchies and Some Operators in OLAP environment*. In: *DOLAP*, pp. 54–59 (1999)
21. Mateus, R.C., Times, V.C., Siqueira, T.L.L., Ciferri, R.R., Ciferri, C.D.A.: *How Does the Spatial Data Redundancy Affect Query Performance in Geographic Data Warehouses?* *JIDM* 1, 519–534 (2010)
22. Brito, J.J., Siqueira, T.L.L., Times, V.C., Ciferri, R.R., de Ciferri, C.D.: *Efficient Processing of Drill-across Queries over Geographic Data Warehouses*. In: *Cuzzocrea, A., Dayal, U. (eds.) DaWaK 2011*. LNCS, vol. 6862, pp. 152–166. Springer, Heidelberg (2011)
23. Brinkhoff, T., Kriegel, H.P., Schneider, R., Seeger, B.: *Multi-step Processing of Spatial*. In: *ACM SIGMOD Conf.*, pp. 197–208 (1994)
24. Mohan, P., Wilson, R., Shekhar, S., George, B., Levine, N., Celik, M.: *Should SDBMS support a join index?: a case study from CrimeStat*. In: *ACM GIS*, pp. 1–10 (2008)
25. Sampaio, M.C., Souza, A.G., Baptista, C.S.: *Towards a Logical Multidimensional Model for Spatial Data Warehousing and OLAP*. In: *DOLAP*, pp. 83–90 (2006)
26. Vaisman, A., Zimányi, E.: *A multidimensional model representing continuous fields in spatial data warehouses*. In: *ACM GIS*, pp. 168–177 (2009)

Measuring the Influence of Built Environment on Walking Behavior: An Accessibility Approach

Guibo Sun, Hui Lin^{*}, and Rongrong Li

Institute of Space and Earth Information Science,
The Chinese University of Hong Kong, Hong Kong, China
{gbsun, huilin, rongrongli}@cuhk.edu.hk

Abstract. Walking behavior has been extensively studied from various perspectives. In this paper, we review the influence of built environment on walking behavior and argue that a longitudinal design with the change of built environment can identify the real influences. We then present a location-based walking accessibility measure for the impact evaluation and describe its methodology with an illustration in a hilly topography community that is experiencing a built environment changes.

Keywords: built environment change, walking behavior, accessibility measure, hilly community.

1 Introduction

Built environment is defined as the composition of urban design, land use, transportation system, and patterns of human activity within this environment [1, 2]. Under the pressures of traffic congestion and climate change, how to design a built environment that can reduce the use of motorized transit and encourage walking and bicycling is important to urban planning and transportation. A challenge for the city movements, such as New Urbanism and Smart Growth, is to identify the relationship between the built environment and walking behavior [3]. The objective and detailed measurement of the built environment that matched with walking behavior is fundamental to the exploration of the impacts of the built environment on walking behavior[4, 5].

Accessibility has been a central concept in urban planning since 1950s, which represents one of the first efforts by planners to develop measures linking land use, transportation and activity systems [6, 7]. It is treated as a generalized indicator to evaluate the built environment change and to relate social and economic impacts [8]. It is reasonable to use accessibility to describe a general picture of the impact of the built environment on walking behavior. However, accessibility measure for walking behavior is rarely involved in previous research[9].

In this paper, the findings on the relationship between built environment and walking behavior from different disciplines are first reviewed. The accessibility approach is then discussed, and a location-based accessibility measure for walking behavior is developed. A distinctive feature of this measure is that it is implemented in a community with high-rises and hilly topography, and can be adapted to cities with dense high-rises.

^{*} Corresponding author.

The rest of the paper is organized as follows: Section 2 reviews the literature on the relationship between built environment and walking behavior. Section 3 describes the proposed accessibility measure using a case study. The concluding remarks and future work are given in Section 4.

2 Built Environment and Walking Behavior

2.1 Built Environment Related to Human Travel

Built environment consists of the following elements: (1) land use, the spatial distribution of buildings and human activities within them; (2) transportation system, the hard transport infrastructure and soft transit service it provides; and (3) urban design, the arrangement and appearance of the physical elements [2]. Research reveals that there exists a relationship between built environment and travel behavior, which is correlated or important. The theoretical foundation can be found in the theory of utilitarian travel demand[10]. Since a majority of trips are derived from the activities one wants to participate in, changing the locations of these activities and/or modifying the design characteristics of the built environment will alter travel patterns [10, 11]. However, inconsistent findings have been obtained due to the differences in research designs (e.g., cross-sectional versus longitudinal [12]), geographical scales (e.g., neighborhoods versus larger regional areas[1, 13]), contexts (e.g., Western cities versus rapidly developing cities [2]), and conceptual and theoretical models (e.g., models with causal relations versus correlation[14-16]).

Walking is regarded as a competitive mode choice with car driving or bus riding. Planners often label a built environment “pedestrian-oriented” if it has relatively high density of development, a mixture of land uses, a street network with high connectivity, human-scale streets, and desirable aesthetic qualities. In this built environment walking will be more viable and appealing than car driving[3]. It should be noted that the relationship between the built environment and walking is different from the one between the built environment and car driving. Psychological and social factors are probably more important for walking than for driving, such as perceptions of safety, comfort, appeal of a streetscape, also the gift of walking time and leisure experience. However, the literature on driving still provides valuable concepts and methods that can be applied to studies of the built environment and walking [1].

2.2 Impacts of Built Environment on Walking Behavior

In attempts to identify environmental influences, a rigorous research design should keep the same individuals within an environment that is subsequently modified, also on the assumption that individual attitudes towards travel would remain stable [15]. Since it is infeasible to change the physical design of a neighborhood, two kinds of compromised methods are usually adopted: (1) by comparing different neighborhood types. The strategy is to examine the differences in walking rates in different environmental characteristics. Often confounding factors, such as the socio-demographic characteristics

of the individuals and neighborhoods, are rarely reported [10, 11]. (2) By finding the respondents who have experienced a change in residential neighborhood [14]. These studies emphasize the exploration of the influence of individuals' attitudes towards transport and possible self-selection of neighborhood. However, if the relocation took place a long time ago, the recall method for the travel behavior and the attitude towards travel would be very vague, moreover, the real reasons behind "self-selection" are less clearly studied [17]. Therefore, longitudinal design with pre-post survey of the changes in walking behavior after the changes of built environment can serve as a better model for the relationship examination.

With respect to the measure for walking behavior, barriers are set up by spatially matching the sufficiently detailed data from built environment with walking behavior [3]. In urban transportation, survey data are usually collected at census collector district (CCD) level [13] with focus on car driving behavior rather than walking behavior, which would inevitably lead to the missing of built environment attribute related with walking. Moreover, aggregated, rather than disaggregated, approaches are usually adopted in the study of the relationship between built environment and walking behavior. Travel behavior is conceptualized in terms of modal choice, travel distance and travel time per trip or even in more aggregated daily travel distance and daily travel time. This problem is more obvious in the studies of public health where walking data is in the form of the frequency, intensity, and duration, while the examination of the built environment variables is rare. More importantly, a specific measure that can evaluate and forecast the impacts of the environmental modifications on actual walking needs to be developed. This measure should be sensitive to the changes of land use and transportation, and be matched with the walking behavior.

2.3 Accessibility and Walking Behavior

Accessibility, a concept used in a number of scientific fields such as transport planning, urban planning and geography, plays an important role in policy making. There are many definitions about accessibility, such as "a measurement of the spatial distribution of activities about a point, adjusted for the ability and the desire of people to overcome spatial separation" [6], "the ease with which any land-use activity can be reached from a location using a particular transport system" [18], and "as the extent to which land-use and transport systems enable individuals to reach activities or destinations by transport modes" [8]. Major research focus is on job-housing accessibility, which is important to the understanding of the urban structure. However, activities such as shopping and recreation in public open space are also believed to be beneficial to the general quality of life. Most accessibility studies mainly focus on automobile-based activities in regional scale. The accessibility for walking behavior has rarely been concerned [9]. In principle, it is logical to measure the accessibility for the walking mode using similar methods to those for motorized vehicle travel [1], thereby allowing policy maker to evaluate the land use and the pedestrian-oriented strategy in neighborhood or community scale. Geographic Information Systems (GIS) can facilitate spatial matching of detailed individual travel behavior data to detailed built environment data. A GIS-based method is thus adopted in this paper.

3 Accessibility Exploring in Controlled Environment

To illustrate our method, the Chinese University of Hong Kong (CUHK) is chosen as a study area. According to Hong Kong Government policy, the CUHK will revert to a four-year program (3-3-4 curriculum) from 2012/13 academic year, which will result in 30 percent increase in its enrollment. Built environment should “supply” more space to meet the increased “demand”. The existing land use and transportation schedule on campus are being adjusted accordingly. All of these will lead to a change in the built environment, which makes the CUHK an ideal controlled laboratory for the exploration of the influence of built environment on walking behavior. The activities concerned in our study are limited to those taking place inside the university boundary so that the socio-demographic of individuals are relatively homogenous. Two travel modes are considered: walking and bus riding (shuttle bus service on campus). It is beneficial to conduct a longitudinal pre-post change survey of the built environment and the travel behavior using the same samples that will have experienced the changes in built environment.

A challenge for this case study is that the CUHK is a community with hilly topography (Fig. 1), which limits vehicular and pedestrian accessibility. The steeply contoured site has resulted in the dispersal of buildings. The precincts are consolidated into clusters of developments, in this way can the benefit of academic interactions be enhanced and the movement between zonal optimized. This mode of development has formed four colleges, each being a congenial neighborhood with its own hostels, dining halls and other facilities. Five more will be established along with the change of built environment. Given this situation, in the current study, the measurement of walking accessibility should be conducted in a three-dimensional (3D) pedestrian network. To the best of our knowledge, studies on the measurement of walking accessibility in a 3D environment are rare. This study makes an attempt in this regard. In the following sections, a detailed pedestrian network is presented and a location-based walking accessibility measure is elaborated.

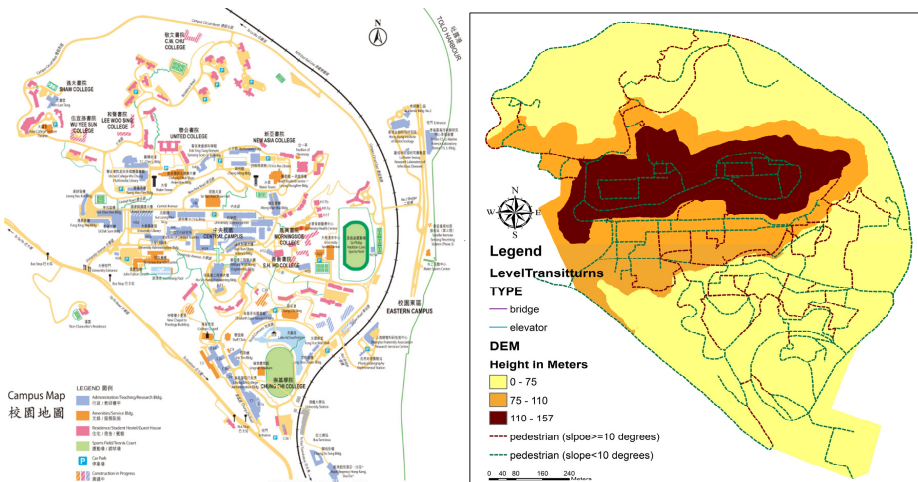


Fig. 1. Left: the study area (campus map of the Chinese University of Hong Kong. Source: <http://www.cuhk.edu.hk/english/documents/campus/cuhk-campus-map.pdf>); Right: three height levels of the CUHK campus ranging from 0 to 156 meters height

3.1 Three-Dimensional (3D) Pedestrian Networks Construction

(a) Detailed Pedestrian Network

The network for walking is quite different from that for car driving. The street network is too coarse to trace the paths chosen by pedestrians. A true pedestrian network should incorporate formal and informal paths, including sidewalks, laneways, pedestrian bridges, and park paths that are informal but frequently used for transportation. The missing pedestrian paths in the street network database are likely to be the ones that are frequently used and can greatly increase the connectivity of separated places in real world. Most studies on accessibility and connectivity use street network only in their analyses, which may cause the inadequacy in description and prediction of travel by walking, hence induce arguments about the reliability of the analysis result [19]. The pedestrian road is usually unavailable in most GIS databases. In the present study, such a dataset was digitized from the map of the CUHK campus, and was supplemented by a field survey.

(b) Barriers on the Connectivity in Hilly Community

Connectivity commonly represents the ease of the travel, and is surrogated by the ratio of the straight line to the real pedestrian road between an origin and a destination. With high connectivity, route distance is similar to straight-line distance. However, in a community with hilly topography, the most important connectivity indicator should probably be the access roads that can overcome the vertical friction to route from this height level to other levels. In this case study, the inconvenience that vertical connectivity imposes on pedestrian movement is reduced with the provision of express lifts. Five express lifts, several foot bridges and hilly stairs on the CUHK campus link various height levels directly (Fig. 2). Similar to the situations in large and fast-growing cities (e.g. Hong Kong), the pedestrian network is experiencing a vertical development and possesses complex 3D topological layouts. Vertical movements in stairwells and elevators, oblique movements on escalators and hilly stairs exist in this hilly community.

(c) A 3D Pedestrian Network Modeling

The necessity of working with 3D network enabled representations of built environments has been demonstrated by [20], which seeks to facilitate effective emergency response. Awareness of the importance of 3D network analysis in small areas, such as in a building, a neighborhood or a community, is proposed to a broader scales of transportation geography and urban studies [21]. Whether which approach is used, the speed, extent, and fidelity of 3D realizations remain somewhat limited [22], for example, 3D topological analytical methods, the 3D shortest route and accessibility analysis functions. But a practical data model can be implemented within a GIS environment without extreme difficulty.

In current study, the outdoor part pedestrian network in two-dimension and a DEM of the CUHK campus were input to the Interpolate Shape (3D Analyst) tool of ArcToolbox, and then a 3D pedestrian network was created. The indoor part, such as

the express lifts, the foot bridges between buildings, and the hilly upstairs were manipulated by Point ZM data model, for example, the express lift was represented using the same x and y location but different z value. Although this 3D relationship is inherently and perceptually intricate, ArcScene makes it easy to visualize such relationship. It forms the foundations of comprehensive 3D network-based accessibility measure. The minimum travel cost between an origin and a destination can then be calculated (Fig. 2).

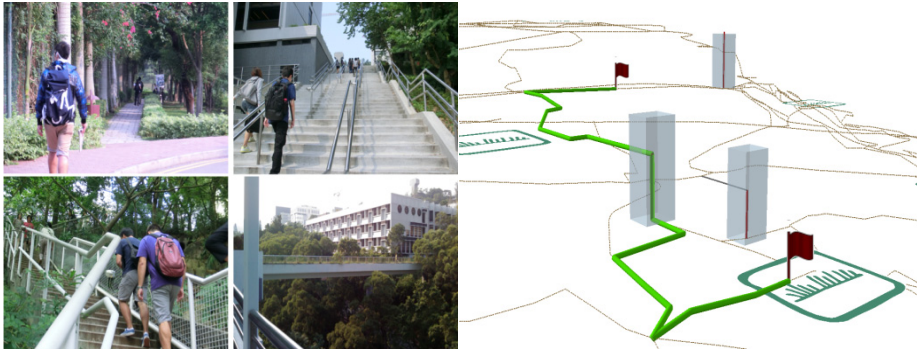


Fig. 2. Left: pedestrian network on campus with hilly topography. Right: routed by 3D pedestrian network in GIS (the green line represents the path with minimum travel cost)

3.2 Walking Accessibility Measure

(a) An Accessibility Measure in Three-dimensional Context

Gravity-based measures have been widely used in urban and geographical studies. A gravity-based measure estimates the accessibility of opportunities in point or zone i to all other n destination zones. The measure can be formulated as follows, assuming a negative exponential cost function:

$$A_i = \sum_{j=1}^n D_j e^{-\beta c_{ij}} \quad (1)$$

where A_i is a measure of accessibility in point (or zone) i to all opportunities D in zone j , c_{ij} the costs of travel between i and j , and β the cost sensitivity parameter. Previous research has suggested that using either time or distance as an impedance variable is acceptable. The negative exponential function is most commonly used as an impedance function. It is also more closely tied with the travel behavior theory [23]. The advantage of this function is that more distant opportunities provide diminishing influences, and thus can provide better estimate for shorter trips, such as those made by non-motorized modes [9, 24].

Major disadvantages of the gravity-based measure are related to the difficult interpretability and communicability as it combines land-use with transport elements, and weighs opportunities [8]. The integral accessibility measure is “defined for a given point as the degree of interconnection with all other points on the same surface” [7]. This measure is easy to be practiced, interpreted and communicated, though it

leaves much to be desired [9]. In this paper, we combine the gravity-based measure with the cumulative opportunities measure [25]. As a result, the accessibility of each location is represented as a combination of transport mode and land use type, and is expressed in decimal indicating the effort needed to travel from an origin to all destinations. The accessibility value calculated for the origin are summed up across all destinations within the fixed acceptable cost value (e.g. 5 minutes walking time), and are normalized by dividing the total number of activities in the study area for the comparison [9] between buildings at the same scale of [0, 1].

To demonstrate our concept and illustrate the procedures, we calculate the accessibility from each building to public open spaces on the CUHK campus. The spatial configurations of public spaces providing for physical recreation, are believed to help increase walking [26]. The accessibility from each building to an open public space on campus is estimated as:

$$A_i = \frac{\sum_{j=1}^m (\text{sizeofPublicSpace}_j * e^{-\beta * \text{TravelCost}_{ij}})}{\sum_{j=1}^n \text{sizeofPublicSpace}_j}, \quad (2)$$

where $n=1,2,\dots,31$ represents the number of public open space; m denotes the number of accessible public open space within the fixed walking time (e.g., 5 minutes), $m \leq n$.

- Step 1: 3D pedestrian network construction. Pedestrian network is detailed in a finer scale and modeled by the methodology in Section 3.1.
- Step 2: Origin-Destination (O-D) Cost Matrix. The travel cost Travel_Cost_{ij} between each O-D pair in the matrix is based on the minimum cost (either time or distance) through the 3D pedestrian network. An origin i is a building, and a destination j a public open space. The impedance (e.g., walking time) value is calculated from the 3D pedestrian network with special treatments of different slopes, considering that walking uphill is difficult than downhill and by express lift.
- Step 3: Accessibility of each building. The size of the i th public open space, $\text{sizeofPublic_Space}_j$ is used to discount the amount of activity opportunities in that destination. The parameter β in Eq. (2) is empirically estimated and a value of 2 is adopted in this paper. In the end, the accessibility of the i th building, A_i , is normalized by the total opportunities of public open space on campus to decimal indicators.
- Step 4: Kriging interpolation for whole community. To get the accessibility of any point within the study area, we interpolate these accessibility values A_i to the whole campus. Accessibilities within 5, 10, 15 and 20 minutes (i.e., different fixed walking time) are shown separately in Fig. 3 for comparison purpose. The accessibility information presented in the figure may assist policy makers in identifying the trend or the influence that might be caused by the re-configuration of facilities.

The O-D cost matrix is calculated using Python programming language. Fig. 3 presents the result plotted in ArcScene v10. It should be noted that the straight line between each OD pair is just a representation of the real 3D shortest pedestrian path. The Kriging interpolation is conducted using the Spatial Analyst Extension, and is visualized in ArcMap v10 (Fig. 4).

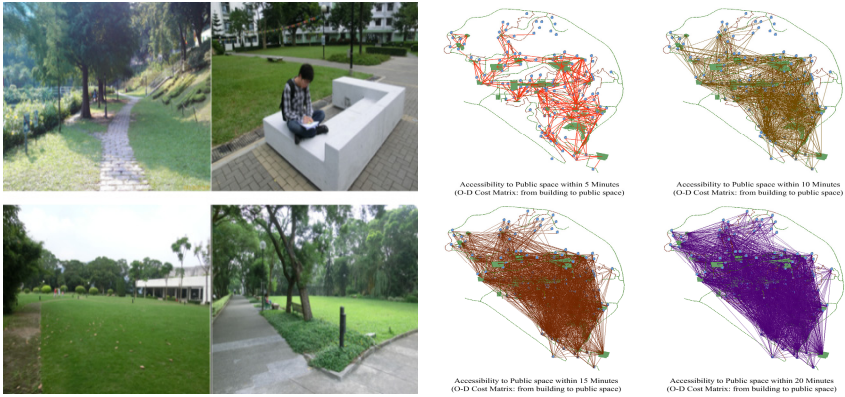


Fig. 3. Left: Public open space on the CUHK campus. Right: O-D cost matrices within 5, 10, 15 and 20 minute walking time, based on the 3D shortest pedestrian path. Obviously, the number of accessible O-D pairs increases with the increase in the fixed walking time.

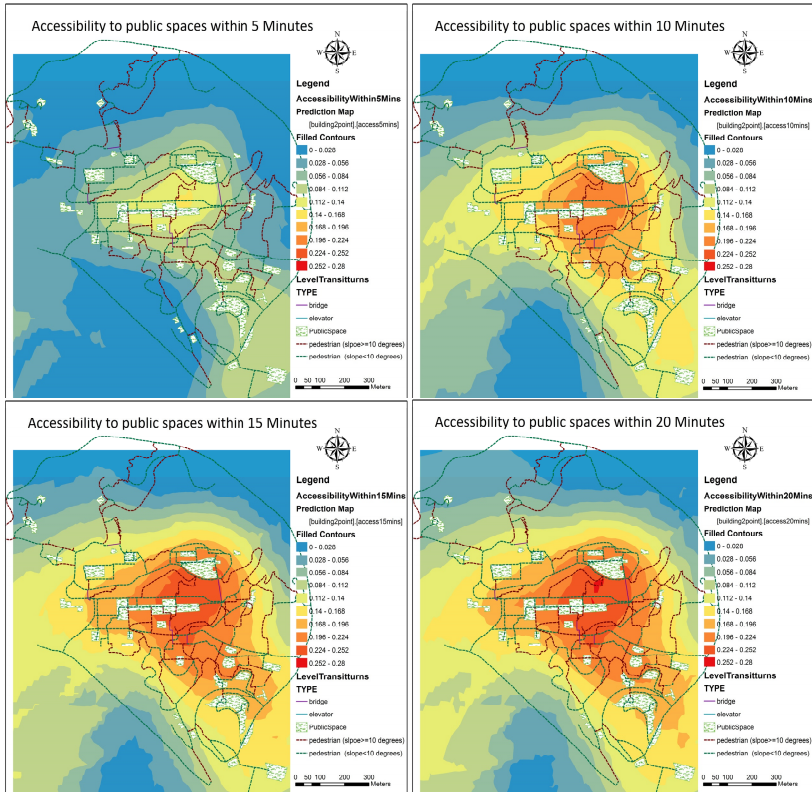


Fig. 4. Maps displaying of accessibility to public space, according to the walking OD cost matrices within 5, 10, 15 and 20 minute walking time. After Kriging interpolation, areas near cluster of public open space (POS) or close to the large size of POS are shown in high accessibility, as one move away from these clusters accessibility gradually decline.

The proposed integral gravity-based accessibility measure evaluates the combined effect of land-use and transportation, and incorporates one's perception on walking by using a distance decay function. It can be easily computed using the existing land-use and transportation data. The capability of visualizing the relative relationship and trend makes the measure interpretable and communicable. In the present study, we merely take the public open space as an example to illustrate the methodology. However, it can be easily extended to other accessibility analyses, such as the residence-shopping accessibility, the spatial configuration of healthcare facilities, especially for community with high-rise and hilly topography.

(b) Generalized Indicator for the Built Environment Change

Accessibility is commonly used in geography to explain the spatial variations, such as population densities, land values and urban growth [7]. However, (1) built environment variables are highly interrelated. With the establishment of a new college (land use change), a new bus stop is settled (transportation changes accordingly), in this situation, individual travel behavior could be influenced directly by the land use change, or indirectly by the alteration of transportation under the land use change. Therefore, an accessibility measure as an integrative indicator combined with the effect of spatial determinants is promising. (2) According to the theory of utilitarian travel demand, accessibility can be reflected by the travel cost. In other words, individuals pay the costs (time or money) to overcome the spatial separation. Shorter distance will encourage the slow mode such as walking. But travel behavior can be forecasted only on the assumption that people would not change their behaved principle after the change of built environment. [10,27] argue that the accessibility benefit will change after the change of environment variables. For instance, a new canteen is opened to increase the accessibility of a local college neighborhood. But due to the tasty of food it supplied, students may choose a further canteen for more tasty-suitable benefits. In this case, our proposed accessibility measure can also be treated as a generalized indicator, and provide quantitative evaluation for policy makers. The estimate of the change combined with the post survey of the travel behavior, a longitudinal experiment design, can then help to identify the relationship between the built environment and walking behavior.

4 Conclusions and Prospects

Walking is one of the most practical means for health improvement. A challenge for urban planners is to identify the relationship between the built environment and walking behavior, and to build the environment beneficial to walking. It is critical to objectively and synthetically measure the built environment. The measurement should be sensitive to land use and transportation change, and is matched with walking behavior. In this paper, a review of the influence of built environment on walking behavior is presented. The significance of longitudinal pre-post survey about the built environment change with the travel behavior to identify the impacts of built environment on walking behavior is illustrated. A location-based accessibility

measure is developed for the CUHK 3D hilly topography community. This measure can be extended to the high-rise sprawl cities, for indoor and outdoor scenarios.

In the present study, we measure the built environment from the location based perspective with an aim of being more objective. As is well known, walking behavior can be influenced by more interpersonal factors. A better understanding of the walking behavior calls for reasonable and effective combination of the Location Theory with Social Psychology. This is our research direction in future.

Acknowledgements. This research is sponsored by Direct Grant from CUHK (CUHK 2021094) and RGC Grant from Hong Kong Research Grants Council (CUHK 405608).

Reference

1. Handy, S.L., Boarnet, M.G., Ewing, R., Killingsworth, R.E.: How the built environment affects physical activity: views from urban planning. *American Journal of Preventive Medicine* 23, 64–73 (2002)
2. Saelens, B.E., Handy, S.L.: Built environment correlates of walking: a review. *Medicine and Science in Sports and Exercise* 40, S550 (2008)
3. Cervero, R., Kockelman, K.: Travel demand and the 3Ds: density, diversity, and design. *Transportation Research Part D: Transport and Environment* 2, 199–219 (1997)
4. Lin, L., Moudon, A.V.: Objective versus subjective measures of the built environment, which are most effective in capturing associations with walking? *Health & Place* 16, 339–348 (2010)
5. Hoehner, C.M., Brennan Ramirez, L.K., Elliott, M.B., Handy, S.L., Brownson, R.C.: Perceived and objective environmental measures and physical activity among urban adults. *American Journal of Preventive Medicine* 28, 105–116 (2005)
6. Hansen, W.G.: How accessibility shapes land use. *Journal of the American Institute of Planners* 25, 73–76 (1959)
7. Ingram, D.: The concept of accessibility: a search for an operational form. *Regional Studies* 5, 101–107 (1971)
8. Geurs, K.T., Van Wee, B.: Accessibility evaluation of land-use and transport strategies: review and research directions. *Journal of Transport Geography* 12, 127–140 (2004)
9. Iacono, M., Krizek, K.J., El-Geneidy, A.: Measuring non-motorized accessibility: issues, alternatives, and execution. *Journal of Transport Geography* 18, 133–140 (2010)
10. Van Wee, B.: Land use and transport: research and policy challenges. *Journal of Transport Geography* 10, 259–271 (2002)
11. Van Acker, V., Witlox, F.: Commuting trips within tours: how is commuting related to land use? *Transportation*, 1–22 (2010)
12. Handy, S.L., Cao, X., Mokhtarian, P.L.: The causal influence of neighborhood design on physical activity within the neighborhood: evidence from Northern California. *American Journal of Health Promotion* 22, 350–358 (2008)
13. Duncan, M.J., Winkler, E., Sugiyama, T., Cerin, E., duToit, L., Leslie, E., Owen, N.: Relationships of land use mix with walking for transport: do land uses and geographical scale matter? *Journal of Urban Health*, 1–14 (2011)

14. Mokhtarian, P.L., Cao, X.: Examining the impacts of residential self-selection on travel behavior: A focus on methodologies. *Transportation Research Part B: Methodological* 42, 204–228 (2008)
15. Saelens, B.E., Sallis, J.F., Frank, L.D.: Environmental correlates of walking and cycling: findings from the transportation, urban design, and planning literatures. *Annals of Behavioral Medicine* 25, 80–91 (2003)
16. Baran, P.K., Rodríguez, D.A., Khattak, A.J.: Space syntax and walking in a new urbanist and suburban neighbourhoods. *Journal of Urban Design* 13, 5–28 (2008)
17. Ewing, R., Cervero, R.: Travel and the built environment. *Journal of the American Planning Association* 76, 265–294 (2010)
18. Dalvi, M.Q., Martin, K.: The measurement of accessibility: some preliminary results. *Transportation* 5, 17–42 (1976)
19. Chin, G.K.W., Van Niel, K.P., Giles-Corti, B., Knuiiman, M.: Accessibility and connectivity in physical activity studies: the impact of missing pedestrian data. *Preventive Medicine* 46, 41–45 (2008)
20. Kwan, M.P., Lee, J.: Emergency response after 9/11: the potential of real-time 3D GIS for quick emergency response in micro-spatial environments. *Computers, Environment and Urban Systems* 29, 93–113 (2005)
21. Lee, J.: A three-dimensional navigable data model to support emergency response in microspatial built-environments. *Annals of the Association of American Geographers* 97, 512–529 (2007)
22. Thill, J.C., Dao, T.H.D., Zhou, Y.: Traveling in the three-dimensional city: applications in route planning, accessibility assessment, location analysis and beyond. *Journal of Transport Geography* 19, 405–421 (2011)
23. Handy, S.L., Niemeier, D.A.: Measuring accessibility: an exploration of issues and alternatives. *Environment and Planning A* 29, 1175–1194 (1997)
24. Kanafani, A.: *Transportation demand analysis*. McGraw-Hill, New York (1983)
25. Kwan, M.P.: Space-time and integral measures of individual accessibility: a comparative analysis using a point-based framework. *Geographical Analysis* 30, 191–216 (1998)
26. Giles-Corti, B., Broomhall, M.H., Knuiiman, M., Collins, C., Douglas, K., Ng, K., Lange, A., Donovan, R.J.: Increasing walking: how important is distance to, attractiveness, and size of public open space? *American Journal of Preventive Medicine* 28, 169–176 (2005)
27. Van Wee, B.: Evaluating the impact of land use on travel behaviour: the environment versus accessibility. *Journal of Transport Geography* 19, 1530–1533 (2011)

Social Welfare to Assess the Global Legibility of a Generalized Map

Guillaume Touya

Laboratoire COGIT, IGN, 73 avenue de Paris, 94165 Saint-Mandé France
Name.surname@ign.fr

Abstract. Cartographic generalization seeks to summarize geographical information from a geo-database to produce a less detailed and readable map. The specifications of a legible map are translated into a set of constraints to guide the generalization process and evaluate it. The global evaluation of the map, or of a part of it, consisting in aggregating all the single constraints satisfactions, is still to tackle for the generalization community. This paper deals with the use of the social welfare theory to handle the aggregation of the single satisfactions on the map level. The social welfare theory deals with the evaluation of the economical global welfare of a society, based on the individual welfare. Different social welfare orderings are adapted to generalization, compared and some are chosen for several generalization use cases. Experiments with topographic maps are carried out to validate the choices.

Keywords: map generalization, evaluation, social welfare, constraints.

1 Introduction

Cartographic generalization is a process that seeks to summarize geographical information from a geo-database in order to produce a less detailed and readable map. Automatic generalization processes were necessary to ease the production of map series and are growingly required nowadays with the development of on-demand mapping. Past research proposed many different approaches to automatically generalize maps [1]. Automatic generalization processes require evaluation procedures both to control and to validate [2]. On the one hand, assessing where the map needs to be generalized is necessary to control which algorithm to use (i.e. enlarge, displace...) and where to use it. On the other hand, an automatic process needs to know if the generalization it performed was successful to validate itself and avoid as much as possible manual post-correction. The automatic evaluation of single specifications in the map, like the minimum area of displayed buildings, has been well tackled [3, 4]. But a map is composed of a very large amount of such specifications and it is very difficult to assess the global quality of a generalized map [3]. Such evaluation is now carried out by human cartographers. Is it possible to aggregate the single evaluations into a unique value that says ‘this map is perfectly generalized according to the specifications’? Is it possible to compare two map generalization alternatives?

This paper tries to answer to these questions, drawing its inspiration from collective welfare theories in economy. Collective welfare assesses the welfare of a human

society from individual welfare [5]. In map generalization evaluation, the specifications for each object can be considered as the individuals in collective welfare.

The second part of the paper deals with the description of the application problem, the global assessment of generalized maps. The third part gives an overview of the collective social welfare theories and describes a benchmark developed to compare social welfare orderings in relation to map specifications satisfaction. The fourth part shows some experiments of our approach inside the automatic generalisation model CollaGen [1]. The last part draws some conclusions and explores further research.

2 Global Assessment of Generalized Maps

As research and technologies allow going further and further towards the automatic generation of maps at different scales, the processes require advanced self evaluation tools. Self evaluation is necessary both to control and validate an automatic process [2]. Indeed, it is necessary to know, at some point of an automatic process, what was well generalized and what is left to do; and as processes success is very dependent on the geographical context they are applied on, it is necessary to rely on self evaluation to validate the result of a process. The point is not to evaluate the quality of generalized data [6], but to answer the following questions: Does the generalized map truly reflect the initial data? Are the transformations due to generalization acceptable? Are the map specifications met? The first part of the section reports related work on generalization evaluation. The second part deals with global assessment from a set of constraints. The third part details three use cases.

2.1 Related Work

Ruas and Mackaness [2] give an overview of past research on the evaluation of automatic map generalization. They state that a generalization evaluation process should include three components: a representation of the real world to verify that generalized data reflect the initial information, a representation of the user's needs (i.e. the expected level of detail of the generalized map and the objectives of the map), and a representation of the rules of cartography and database integrity.

A first method is to identify the geographic characters that bear the user's needs and the cartographic rules, and to assess if generalization is successful according to each character, thanks to spatial measures. For instance, it consists in measuring if the buildings are large enough or if the roads coalesce. Similarly, in the AGENT automatic generalization model [7], each object is able to evaluate its characters in relation to the specifications. In order to benchmark existing generalization algorithms, a template for sharing the measures to evaluate the most common characters of geographic objects was proposed [8]. More recent research went further in this approach, defining evolution function for each of the character [3]. Evolution function give the expected final value of a character according to the initial value and thresholds extracted from the specifications. Fig. 1 shows an example of evolution function for the area of buildings. The evolution functions can be determined using cartographers knowledge or by reverse engineering [4]. Experiments show that such

an approach allows evaluating individually every element of the user's needs, the cartographic rules and even the respect of the initial data [4]. Nevertheless, difficulties remain in the evaluation of characters of complex group of objects like building alignments [9].

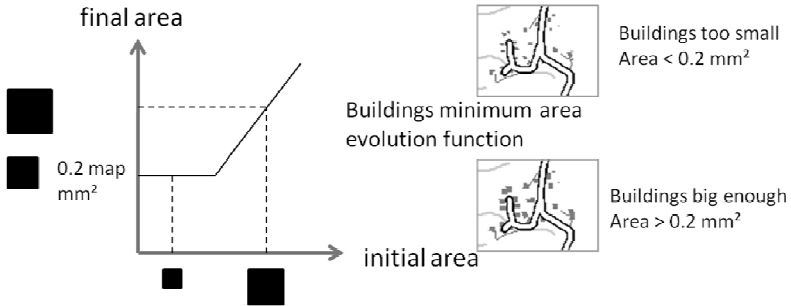


Fig. 1. Evolution function of a constraint on building minimum area

Although it allows a fine description of the legibility of a generalized map, the previous approach only provides the building blocks for a global evaluation of the map. The aggregation of these elementary evaluations is stated as an important issue: a simple solution is to express the elementary evaluation in unique Likert-like scale [10] and to compute the mean (or median) of the elementary evaluations, weighted by the relative importance of the characters [1, 3, 7]. Without satisfying automation, past research resorted to human cartographers' evaluation, using ranking templates [8, 4].

Relying on human intervention is prohibited in automatic processing, so it is necessary to figure out how to globally assess a generalized map. The next section describes the framework built to study the global evaluation and why the simple mean of elementary evaluations is not sufficient.

2.2 Approach: Global Assessment from a Set of Monitors

In order to globally assess generalized maps, we propose a framework that benefits from the past research on the elementary evaluation of characters. In this framework, constraints are used to represent the user's needs and the cartography rules. Constraints have proved to be the most relevant way to model the specifications [11] and make consensus in the generalization research community to serve as input of generalization processes [1, 4]. A constraint constrains the value of character of a map object, once generalized. (C1): "Buildings area should be bigger than 0.2 map mm²" and (C2): "Roads initial general shape should be maintained" are two examples of generalization constraints. To ensure a representation of the real world [2], a geographical objects ontology can be used to express the constraints [1].

In the proposed framework, the constraints that represent user's needs are monitored by constraint monitors like in [1]. Constraints monitors locally convey the evaluation of the constraint satisfaction for a given object. As suggested in [7], constraint monitors satisfaction is comprised in an integer interval, between 1 and 8.

The integer values make comparisons and interpretation easier, compared to real values. The number of possible values was five in [7] but is extended here to eight in order to make the expression of small improvements easier. Qualitative descriptions are associated to the satisfaction value as abstraction of measures helps the understanding of the value [12] (e.g. 1 is represented by “Unacceptable” and 8 by “Perfect”): the satisfaction scale is thus Likert-like [10]. In order to compute its satisfaction, each monitor is endowed with its own method that compares current and goal values. For instance, (C1) is monitored by as many monitors as buildings in the map (e.g. if there are 500 buildings in the map, 500 monitors monitor (C1)), and the monitor uses the evolution function from Fig. 1 to assess satisfaction: if the current building area is equal to the goal area of the evolution function, satisfaction is “Perfect” (i.e. 8), if the area is bigger than 60% of goal area, satisfaction is “Fair” (i.e. 4), etc.

In this framework, the global assessment of generalized maps comes to evaluate the distribution of monitors’ satisfaction. The utilitarian method (i.e. using the satisfactions mean [13]) as proposed by [1, 3, 7] is not enough to assess globally a generalized map. Fig. 2 helps to illustrate this statement: two generalization alternatives are evaluated with nine monitors that monitor constraints on proximity between buildings and proximity between building and road symbols. If the mean of satisfaction is used, the n°2 is preferred (7.2 against 5.8) whereas a human cartographer would prefer n°1, especially for a direct printing of the map, as no big legibility problem remain.

The constraints do not all have the same importance in a generalized map [4,7,11] (e.g. minimal size is more important than preserving initial position), so the satisfaction distributions are weighted by constraint importance. But for the sake of simplicity, in this paper, all constraints have equal importance.

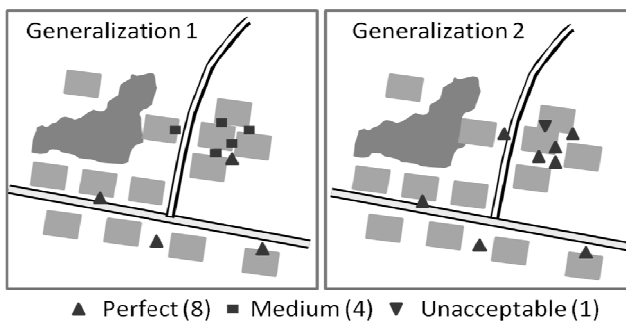


Fig. 2. Two generalization solutions evaluated by a sample of the constraint monitors

2.3 Use Cases to Distinguish

The global assessment of a generalized map (or of a generalized part of a map) can be necessary in several situations, when generalization is carried out by automatic processes. Thus, three use cases, where global assessment is required, have been

identified: evaluating a final output (ready to print), evaluating a final output with possible manual post-editing and comparing the legibility of the map with the previous step in an iterative automatic process.

The first use case is the evaluation of a final output, a map ready to be printed without further correction of remaining legibility problems. In this use case, major legibility problems that blur map interpretation should be avoided. Considering distributions, it means that constraint satisfaction distributions with the more perfect satisfaction are preferred as the perfect satisfaction is the only one that guarantees no legibility inconvenience. Moreover, distributions with less unacceptable satisfactions are preferred as such satisfactions are supposed to represent very inconvenient legibility problems while the medium satisfactions represent minor problems that do not disturb global legibility much. In this use case, in Fig. 2, (1) would be preferred to (2).

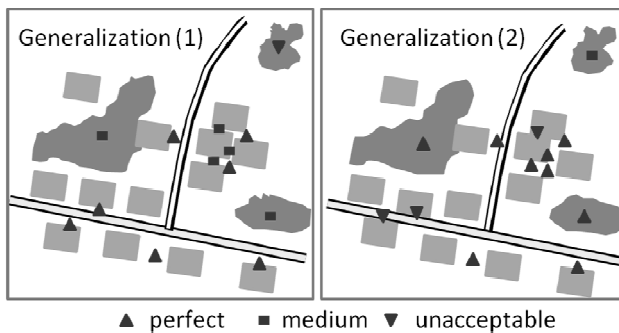


Fig. 3. Two generalizations that illustrate the second use case: manual editing. N°2 is preferred to n°1 as fewer manual editing are required in this solution.

The second use case is the evaluation of a final output of an automatic process but with possible manual editing of the remaining legibility problems. This use case is very important as most of actual map production lines based on automatic generalization resort to manual post-editing of the automatically produced map. This case is the opposite of the previous one as distributions with marked differences in satisfaction values and less medium values are preferred. Indeed, all imperfect satisfaction monitors (even medium ones) require manual edition of the corresponding legibility problem. Moreover, the correction of a legibility problem identified by medium satisfaction is as long as a big legibility problem. Thus, in the example of Fig. 3, n°2 is preferred as it only requires 4 manual edits (3 unacceptable satisfactions and 1 medium one) while 6 are required for n°1 (1 unacceptable satisfaction but 5 medium ones).

The last use case is the evaluation of a positive evolution of satisfaction compared to the previous state of the map in an iterative automatic process. Contrary to the second use case, the global legibility of the map is not assessed but only the global evolution of satisfactions. In this case, the distributions with less low satisfactions and more high satisfactions are preferred, whatever the evolution on medium satisfaction is.

We believe that the simple mean of the satisfactions is not enough to evaluate globally a generalized map in the three use cases, and that collective welfare theories

may provide alternatives. In order to prove the assertion, we carried out a four step reasoning: first implement a library of social welfare orderings (i.e. potential alternatives to mean) from literature; then define a set of toy distributions that illustrate the diversity of constraints distributions; analyze how social welfare alternatives order the toy distributions to choose one adapted to each use case; finally, prove choice validity by testing on real generalized data. The next section introduces the theories of collective welfare, and describes the first three steps of our reasoning.

3 Theories of Collective Social Welfare

3.1 Collective Welfare and Social Welfare Ordering

Microeconomic analysis and economic theories assume that each individual tries to maximize its own preferences, its welfare, often assimilated to the money earned by the individual. Then, collective welfare is the aggregation of the individual welfare of every member of a society, and economical policies intend to maximize collective welfare [5]. We assume that the global assessment of generalized maps can be considered as a collective welfare problem where the individuals are the constraint monitors and their utility is their satisfaction. Cardinal welfarism studies social welfare orderings (SWOs) in order to analyze the collective welfare of a society. SWOs allow comparing two societies according to collective welfare, using the individual welfare. Different SWOs may convey different perspectives on collective welfare favoring either the total society welfare or the decrease of welfare difference between individuals. For a given individual utilities distribution $(u_1, u_2 \dots u_n)$ noted (u_i) and a second distribution (u'_i) , a social welfare ordering provides an order relation, i.e. allows to assess that (u_i) is preferred to (u'_i) , which can be noted:

$$(u_i) > (u'_i) \quad (1)$$

A great diversity of social welfare orderings can be defined to compare distributions of individual utility. For instance, the utilitarian social welfare ordering, coming from Bentham philosophy [13], considers the sum of individual utilities to compare collective distributions (Equation 2).

$$(u_i) > (u'_i) \Leftrightarrow \sum u_i > \sum u'_i \quad (2)$$

A Collective Utility Function (CUF) may be related to a social welfare ordering. The CUF gives a real value representing the value of the collective welfare according to the scale of utilities. For instance, the collective utility function related to the utilitarian social welfare ordering of Equation 2 is the mean of the individual utilities. SWOs and their CUF are not exclusively used in economy but also in computer science and particularly multi-agents systems research in order to solve resource allocation problems that require fair division of resources [14].

Historically, there are three approaches to collective welfare: the utilitarian, the egalitarian and the Nash orderings. The utilitarian orderings introduced above, consider the sum of individual utilities as the basis for SWOs and related CUF. Such

an approach will prefer a society with equally distributed unsatisfied individuals and very satisfied individuals, compared to one where all the individual utilities are equal to the mean. Egalitarian orderings follow the justice principles of Rawls [15]. In order to favor fair distributions, egalitarian orderings will compare the least satisfied individuals [5]. Finally, Nash orderings try to balance utilitarian and egalitarian approaches by using the product of individual utilities as the basis for ordering. Nash orderings penalize the distributions with some very unsatisfied individuals.

We developed a library of social welfare orderings, following the three approaches, which convey different behaviours to compare distributions. Within this quite large library, choices can be made to use the most appropriate ordering in order to compare global generalisation results in the proposed use cases. These social welfare orderings and their CUF are described in the next section.

3.2 A Library of Social Welfare Orderings

The social welfare orderings described in this section are illustrated with a simple example composed of three distributions of individual utilities, u , v and w (Equation 3). There are five individuals and their utility varies from 1 to 8.

$$u = \{1,2,8,8,8\}, v = \{2,3,6,6,6\}, w = \{4,4,4,4,4\} \tag{3}$$

Utilitarian SWOs. The utilitarian SWOs derive from the classical utilitarian SWO [13] presented in Equation 2: utilities are summed. The utilitarian SWO ranks the u distribution as the best, as the utilities sum is the highest (27 against 23 for v and 20 for w). A simple alternative to the utilitarian SWO is a powered utilitarian SWO (Equation 4) that penalizes the low utilities when the power parameter is high.

$$(u_i) \succ (u'_i) \Leftrightarrow (\sum u_i^p)^{1/p} > (\sum u'_i{}^p)^{1/p} \tag{4}$$

The Iso-Elastic SWO is a more egalitarian version of utilitarianism when its parameter p tends to infinity (Equation 5). For instance, with 5 as a parameter value, the examples are ranked differently: $w > v > u$.

$$(u_i) \succ (u'_i) \Leftrightarrow (\frac{1}{1-a} \sum u_i^{1-a}) > (\frac{1}{1-a} \sum u'_i{}^{1-a}) \text{ where } a \in [0,1[\cup]1, +\infty] \tag{5}$$

The Owa SWOs are a family of orderings that weight each individual utility differently [16]. Depending on the weights assigned to low, mean or high individual utilities, an Owa SWO may derive from utilitarian SWOs and favour some specific distributions (Equation 6).

$$\text{for a function } w(u_i), (u_i) \succ (u'_i) \Leftrightarrow \sum w(u_i) \cdot u_i > \sum w(u'_i) \cdot u'_i \tag{6}$$

Egalitarian SWOs. The egalitarian SWOs follow Rawls principles of justice [15], penalizing the distributions with a low minimum. Most egalitarian SWOs rely on the Leximin order [17]. Equation 7 represents the Leximin order: the (u_i) distribution sorted by ascending order is noted $(u_i)^*$. The equation means that distributions are

compared by their number of low utilities. With this SWO, the rank of the example distributions is different than the utilitarian ones as $w > v > u$ (Fig. 4).

$$(u_i) > (u'_i) \Leftrightarrow \exists i \in \llbracket 1, n - 1 \rrbracket / (\forall j \in \llbracket 1, i \rrbracket, u_j^* = u_j'^*) \wedge u_{i+1}^* > u_{i+1}'^* \quad (7)$$

The Leximin SWO is very strict and alternatives have been developed to weaken its effects while keeping an egalitarian ordering [18]. For instance, a poverty line can be introduced: for a poverty line at utility 2, the vectors u and v are ordered using the Leximin principle because the lowest different values are below the threshold, while vectors like w are compared on an utilitarian basis as all values are over the poverty line. The poverty line may convey the idea that only the very low individual utilities are unacceptable, which can be useful for our generalization evaluation purpose.

It is worth noting that Leximin-based SWOs, do not have a related CUF, able to help differentiate distributions. As a consequence, the egalitarian SWOs will not be useful in use cases where a CUF is needed (use case 1 and 2).

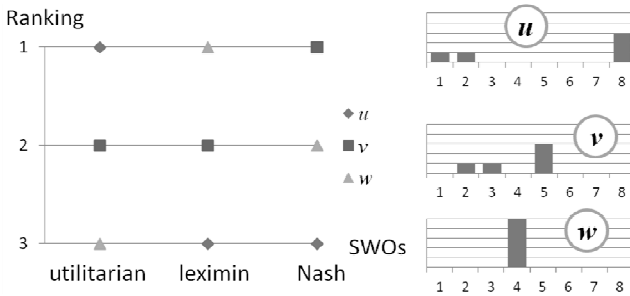


Fig. 4. Summary of the examples distribution orders for the three main SWOs

Nash SWOs. Finally, the Nash SWOs multiply all individual utilities (Equation 8). Here, the rank of the example distributions is different as $v > w > u$ (Fig. 4).

$$(u_i) > (u'_i) \Leftrightarrow \prod u_i > \prod u'_i \quad (8)$$

The Nash welfare can be derived to a powered Nash welfare, to favor big utilities (Equation 9). With the powered Nash welfare, the gap between v and w increases, while the gap between w and u decreases.

$$(u_i) > (u'_i) \Leftrightarrow (\prod u_i^p)^{1/p} > (\prod u_i'^p)^{1/p} \quad (9)$$

3.3 Social Welfare Ordering Benchmarking

In order to select the SWOs that could be useful to improve generalization global evaluation in several use cases, it is necessary to comprehend how the SWOs presented in the previous section behave to sort standard distributions. If it is possible to know that a particular SWO favors more than the others a given distribution, this

SWO could be used to evaluate generalization in use cases where such a distribution is preferred. This section describes a SWO benchmarking that was carried out to meet this objective, based on toy distributions.

Toy Distributions of Constraint Satisfaction. Toy distributions are distributions of one hundred monitors with specific patterns. The “Medium” toy distribution is composed of 50 monitors with *fair* satisfaction (4) and 50 with *acceptable* satisfaction (5), while the “Extreme very good” toy distribution is composed of 30 monitors with satisfaction 1 and 70 with satisfaction 8 (Fig. 5). Considering social welfare, each of the hundred individuals of our toy distribution has a utility that may vary from 1 to 8. Besides, the 11 toy distributions allow verifying that SWOs do not mix up big trends: the “very good” distributions should always be preferred to medium ones.

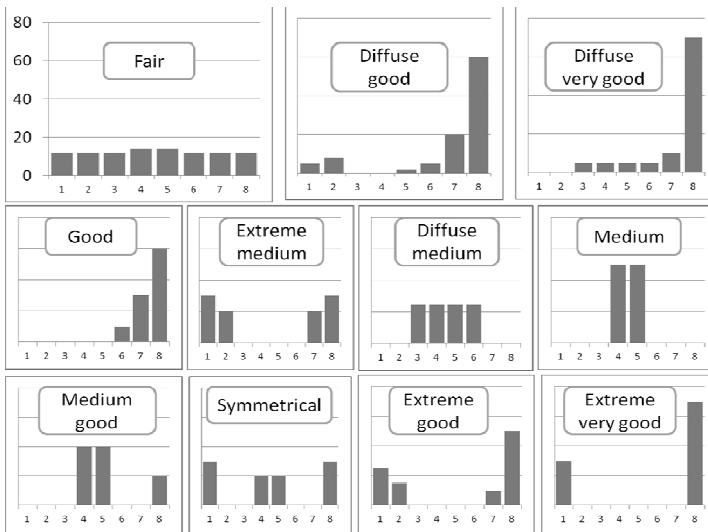


Fig. 5. Utilities histogram (100 individuals) of all the toy distributions

Ranking the Toy Distributions. Experiments were carried out to benchmark a large set of SWOs with the toy distributions, as the way a SWO ranks the toy distributions illustrates its future behavior with monitor satisfaction distributions. The benchmarked SWOs are thirty ordering methods obtained from the equations previously described, with several parameters for the SWOs that can be parameterized (and also other SWOs from the literature). For each SWO, the result is a list of the toy distributions ordered by preference. For instance, the utilitarian SWO prefers the “Good” toy distribution, then “diffuse very good”, “diffuse good”, “extreme very good”, “extreme good”, “medium good”, “symmetrical”, “medium”, “diffuse medium”, “extreme medium” and finally the “fair” distribution. Two different analyses were carried out on the ranking lists: a comparison to the mean ordering of each toy distribution and a comparison to the utilitarian list.

The comparison to the mean ordering is computed for each SWO: it is the difference, for each toy distribution (e.g. “fair”), between the mean rank of the

distribution in the sorted list of all SWOs (9th out of 11 for “fair”) and the rank of the toy distribution in the sorted list of a given SWO (11th for “fair” for the utilitarian SWO). This comparison allows finding the favored distributions and the ones that are penalized by a given SWO. For instance, it allows knowing that the powered utilitarian SWO greatly favors more the “medium extreme” distribution (Fig. 5) than the others.

Table 1. Comparison in toy distribution ranking between standard utilitarian method and a sample of the alternative ones

Evaluation Method	FAIR	GOOD LOOSE	VERY GOOD LOOSE	GOOD	MEDIUM EXTREME	MEDIUM LOOSE	MEDIUM	MEDIUM GOOD	SYMMET RICAL	GOOD EXTREME	VERY GOOD EXTREME
StandardUtilitarianMethod	0	8	9	10	1	2	3	5	4	6	7
PoweredUtilitarianMethod (5.0)	2	0	1	-1	4	-1	-3	-2	0	0	0
LeximinPovertyLine (3.0)	4	-3	0	0	-1	4	4	3	-3	-3	-5
WeakPovertyMean (2.0, 6.0)	4	-3	0	0	-1	4	4	3	-3	-4	-4
OwaWelfare (4, 3, 2, 1, 1, 2, 3, 4)	3	0	1	-1	4	-1	-3	-3	0	0	0
OwaWelfare (1, 1, 1, 4, 4, 1, 1, 1)	4	-5	-3	-5	-1	6	7	4	3	-5	-5
OwaWelfare (3, 3, 3, 2, 2, 1, 1, 1)	4	-3	-1	-3	-1	4	6	5	-2	-5	-4
IsoElasticMethod (30.0)	2	-3	0	0	-1	4	4	3	-3	-3	-3
IsoElasticMethod (0.5)	2	0	0	0	-1	2	3	2	-3	-3	-2
IsoElasticMethod (0.2)	2	0	0	0	-1	1	1	1	-3	-1	0
NashWelfare	2	0	0	0	-1	3	3	2	-3	-3	-3
BernoulliNashWelfare	4	0	0	0	-1	3	3	2	-3	-4	-4
AtkinsonWelfare (0.2)	4	0	0	0	-1	3	3	2	-3	-4	-4
AtkinsonWelfare (-10.0)	2	-3	0	0	-1	4	4	3	-3	-3	-3

The second analysis compares the ranks of every toy distribution in a given SWO preference list to the standard utilitarian preference list ranks (as it the standard method actually used in generalization). The comparison is summarized in Table 1 for a sample of the tested SWOs. This analysis allows finding the distributions that are favored and the penalized ones, by a given SWO compared to the utilitarian SWO. Therefore, the analysis highlights the SWOs that could be used in the use cases where utilitarianism fails. It can be noted that the non-utilitarian SWOs can be divided into three groups: the egalitarian, the SWOs balanced between egalitarianism and utilitarianism (that include Nash, iso-elastic and Atkinson SWOs) and the Owa SWOs that have very unique behaviors depending on their weight parameters. Thanks to both analyses, the next part proposes to match the use cases with the most appropriate SWO.

Relating Social Welfare Orderings to Use Cases. The way the SWOs rank the toy distributions allow to infer their behavior against the four use cases described in section 2. Therefore, it helps us to assign a SWO to use in each use case, in order to improve the current utilitarian methods. The first one is the evaluation of a final output map, ready to be printed on paper or on a screen. According to Table 1, only two SWOs favor the “diffuse very good” distribution over the “good” one: the powered utilitarian SWO and the Owa SWO with high weights on extreme satisfactions. As the Owa SWO favors more the “fair” distribution, it is chosen to assess the final outputs.

The second use case is the evaluation of a map with possibilities of manual post-editing. This use case requires to favor distribution with the fewest possible unsatisfied constraints, whether the dissatisfaction is severe or just medium. The utilitarian SWO is quite satisfying for the use but a powered utilitarianism with a high power (e.g. 5, Table 1) is much better as it penalizes the “medium” or “diffuse medium” distributions that have many medium satisfactions (i.e. that require heavy post-editing).

The final use case is the evaluation of a map compared to the previous state of the map. The use case requires a SWO that conveys improvements in its preferences: fewer low utilities and more high utilities. Moreover, a CUF is not necessary as distribution are only compared to the previous one, so all leximin-based SWOs, that do not have a CUF, can be used in this use case. No single SWO seems to be sufficient in this use case, so we propose to use a leximin with poverty line SWO to measure the decrease of low utilities and an iso-elastic SWO with a parameter value close to 0 (e.g. 10^{-5} , see Table 1) to measure the increase of high utilities.

In the next section, experimentations validate the choices made for each use case.

4 Application to CollaGen Model Generalisations

The CollaGen model is an automatic framework to carry out iterative generalization procedures that use different existing processes on different parts of the map [1] (Fig. 6). For instance, the cities are first generalized by process 1, then, rural areas by process 2 and finally roads by process 3. In CollaGen, the three use cases can occur, so it is possible to experiment the social welfare evaluation framework. In the experiment, a sample with 25 constraints is used including constraints on one object (building minimum area, maintain road initial shape, etc.), on two objects (e.g. minimum distance between buildings) and on group of objects (density of building blocks, maintain building alignments, etc.). For instance, the 25 constraints are managed by 13895 monitors in the map city of Fig. 6.



Fig. 6. A generalized city, whose satisfaction distribution contains nearly 14000 monitors

Use case 1 is illustrated by the final output of a 1:50k map, generalized with the CollaGen model (Fig. 7). As the results are very good, the distribution is mainly composed of very satisfied constraints and the Owa welfare SWO gives a value of 5.23 while the utilitarian SWO gives 5.72. The generalization result is slightly

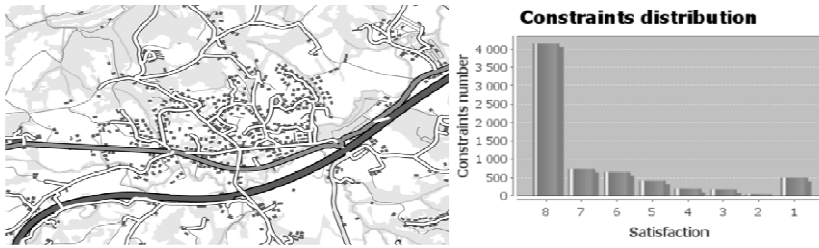


Fig. 7. Use case 1: a very good final output at 1:50k, obtained with CollaGen [1]. This kind of distributions is better comprehended by an Owa SWO.

damaged artificially on the medium satisfied constraints, to simulate a less good process: the results confirm that Owa welfare is less sensitive to variations in medium satisfactions as they show a bigger decrease in the utilitarian CUF than in the Owa one. Moreover, another alternative generalization is carried out by damaging a few maximum satisfactions that become minimum satisfactions. The experiment shows that the Owa SWO is more sensitive to that, with a bigger decrease of the CUF (0.9 against 0.4). Therefore, Owa seems to be well adapted to the use case 1.

Use case 2 (final output with manual editing) is illustrated by a rural generalized map by two automatic processes, the second one averaging the constraint satisfaction, which should be penalized by the chosen SWO (Fig. 8). Utilitarian welfare ranks both generalizations the same way (5.70) so it is well adapted to use case 2. On the other hand, powered utilitarian welfare ranks generalization (1) first with a 5.73 CUF against 5.68 for generalization (2), showing its better accommodation to use case 3.

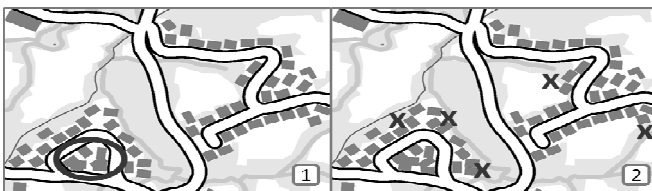


Fig. 8. (1) good for manual editing with few (but major) conflicts in the circled area (2) optimizing generalization, not suited for manual editing (crosses show required minor edits)

Use case 3 is illustrated by the generalization of a city with several iterative processes. The first process generalizes most of the city, then, two alternative processes are used and evaluated with the previous state to assess a significant improvement (Fig. 9). The experiments showed that the leximin with poverty line SWO chosen to assess decreases in low satisfactions is not enough as all generalization results, and even the good ones, remain partly unsatisfied, which limits the differentiations. So, the SWO was adapted to use the quantity of constraints under the poverty line to compare the distributions. Such a change produces good result: while the utilitarian SWO ranks the improvement exactly the same (5.6 against 5.0 for the previous distribution), the adapted leximin SWO considers (D3) as a significant improvement (6.9% decrease in unsatisfied constraints)

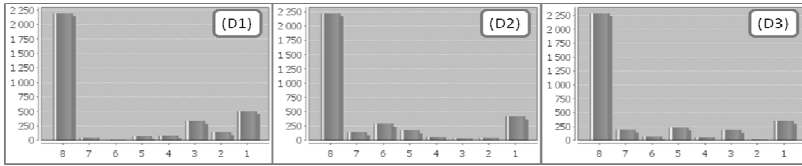


Fig. 9. (1) Distribution after first process. (2) Distribution after alternative process 1: negligible improvement. (3) Distribution after alternative process 2: it is a significant improvement.

and (D2) as negligible (2.4% decrease). The difference is weakened considering the high satisfactions, which is well translated by the Iso SWO that considers the improvement as a bit better in (D3) as in (D2).

Do the previous applications prove that collective welfare theories significantly improve generalization global evaluation? Several points can be discussed. First, the application of different SWOs to the three use cases shows the advantage of such a vast theory, i.e. its flexibility: we found at least one SWO that fits better than mean, with the purposes of each use case. Moreover, an alternative would be to use multiple criteria decision techniques [19], each constraint being a criterion. But the large number of monitors in a map (Fig. 6) better fits the number of individuals in a society, as multiple criteria decision uses fewer criteria (5 to 15 in most techniques). Another advantage of collective welfare analogy is its expressiveness. For instance, the use of a poverty line in use case 3 is quite expressive. Finally, although differences are not huge, the applications show the sensitivity allowed by collective welfare: it is possible to deal with use cases where mean is not able to give completely satisfying result.

However, there are a lot of SWOs to discriminate. So, it requires intensive testing (e.g. with toy distributions) to find the best SWO for a given use case.

5 Conclusion and Further Work

To conclude, we proposed a framework to assess the global legibility of a generalized map, based on a parallel with collective welfare theories. Individual constraints that monitor legibility are considered as individuals in a collective society with a level of satisfaction. In the economical field, several methods (Social Welfare Orderings) exist to measure the global welfare of the society from individual welfare. Several SWOs have been compared in relation to the global evaluation of generalized maps and some have been chosen to meet some use cases demands. The experiments, carried out on generalized topographic maps, validate the use of collective welfare methods.

To go further, other generalization use cases could be tested, where other SWOs would provide better results than the mean. Moreover the SWOs could be used to assess, on the same data, automatically and manually generalized maps, used as references. It would allow assessing if constraints really convey all specifications, without the softening effect of the mean. Then, the chosen SWOs could be tuned to be even more adapted to the use cases: machine learning on result samples could help to weight the SWOs like in [20]. Finally, it would be interesting to use SWOs to solve the legibility problem of geoportals, studied in [21].

References

1. Touya, G., Duchêne, C.: CollaGen: Collaboration between automatic cartographic generalisation processes. In: Ruas, A. (ed.) *Advances in Cartography and GIScience*, pp. 541–558. Springer, Berlin (2011)
2. Mackaness, W.A., Ruas, A.: Evaluation in the map generalisation process. In: Mackaness, W.A., Ruas, A., Sarjakoski, L.T. (eds.) *Generalisation of Geographic Information*, pp. 89–111. Elsevier, London (2007)
3. Bard, S.: Quality assessment of cartographic generalisation. *Transactions in GIS* 8(1), 63–81 (2004)
4. Stoter, J., Burghardt, D., Duchêne, C., Baella, B., Bakker, N., Blok, C., Pla, M., Regnauld, N., Touya, G., Schmid, S.: Methodology for evaluating automated map generalization in commercial software. *Computers, Environment and Urban Systems* 33(5), 311–324 (2009)
5. Moulin, H.: *Fair Division and Collective Welfare*. The MIT Press (2004)
6. Goodchild, M.F., Jeansoulin, R. (eds.): *Data Quality in Geographic Information: From Errors to Uncertainty*. Hermes, Paris (1998)
7. Ruas, A.: The roles of meso objects for generalisation. In: *Proceedings of 9th International Symposium on Spatial Data Handling*, Beijing, China, vol. 3b, pp. 50–63 (2000)
8. Ruas, A.: Automatic generalisation project: Learning process from interactive generalisation. *OEEPE Official Publication* 39 (2001)
9. Zhang, X., Stoter, J., Ai, T., Kraak, M.-J.: Formalization and data enrichment for automated evaluation of building pattern preservation. In: Guilbert, E., Lees, B., Leung, Y. (eds.) *Proceedings of Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science*. ISPRS archives, vol. XXXVIII (2010)
10. Likert, R.: A technique for the measurement of attitudes. *Archives of Psychology* 22(140), 1–55 (1932)
11. Beard, K.: Constraints on rule formation. In: Buttenfield, B., McMaster, R. (eds.) *Map Generalization*, pp. 121–135. Longman, New York (1991)
12. Mustière, S., Zucker, J.-D., Saitta, L.: Abstraction-Based machine learning approach to cartographic generalisation. In: *Proceedings of 9th International Symposium on Spatial Data Handling*, Beijing, China, vol. 1a, pp. 50–63 (2000)
13. Bentham, J.: *An Introduction to the Principles of Morals and Legislation*. Clarendon Press, Oxford (1789)
14. Nongaillard, A., Mathieu, P., Jaumard, B.: A Realistic Approach to Solve the Nash Welfare. In: Demazeau, Y., Pavón, J., Corchado, J.M., Bajo, J. (eds.) *7th International Conference on PAAMS 2009*. AISC, vol. 55, pp. 374–382. Springer, Heidelberg (2009)
15. Rawls, J.: *A theory of Justice*. Belknap, Cambridge (1971)
16. Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Trans. Syst. Man Cybern.* 18, 183–190 (1988)
17. Sen, A.: Rawls versus bentham: An axiomatic examination of the pure distribution problem. *Theory and Decision* 4(3), 301–309 (1974)
18. Tungodden, B.: Egalitarianism: Is leximin the only option? *Economics and Philosophy* 16(2), 229–245 (2000)
19. Figueira, J., Greco, S., Ehrogott, M. (eds.): *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer (2005)
20. Taillandier, P., Gaffuri, J.: Designing generalisation evaluation function through human-machine dialogue. In: *Proceedings of GIScience 2010*, Zurich (2010)
21. Stigmar, H., Harrie, L.: Evaluation of analytical measures of map legibility. *The Cartographic Journal* 48(1), 41–53 (2011)

Investigations into the Cognitive Conceptualization and Similarity Assessment of Spatial Scenes^{*}

Jan Oliver Wallgrün, Jinlong Yang, Alexander Klippel, and Frank Dylla

Department of Geography, GeoVISTA Center, The Pennsylvania State University
{wallgrun, jinlong, klippel}@psu.edu
Cognitive Systems, Spatial Cognition SFB/TR 8, Universität Bremen
dylla@sfbtr8.uni-bremen.de

Abstract. Formally capturing spatial semantics is a challenging and still largely unsolved research endeavor. Qualitative spatial calculi such as RCC-8 and the 9-Intersection model have been employed to capture humans' commonsense understanding of spatial relations, for instance, in information retrieval approaches. The bridge between commonsense and formal semantics of spatial relations is established using similarities which are, on a qualitative level, typically formalized using the notion of *conceptual neighborhoods*. While behavioral studies have been carried out on relations between two entities, both static and dynamic, similar experimental work on complex scenes involving three or more entities is still missing. We address this gap by reporting on three experiments on the category construction of spatial scenes involving three entities in three different semantic domains. To reveal the conceptualization of complex spatial scenes, we developed a number of analysis methods. Our results show clearly that (I) categorization of relations in static scenarios is less dependent on domain semantics than in dynamically changing scenarios, that (II) RCC-5 is preferred over RCC-8, and (III) that the complexity of a scene is broken down by selecting a main reference entity.

1 Introduction

Formally capturing spatial semantics is a challenging and still largely unsolved research endeavor. Over the last two decades, a multitude of different spatial (and temporal) formalisms, often referred to as qualitative spatial calculi, have been suggested in the literature to model human commonsense understanding of spatial and spatio-temporal relations (see Cohn & Renz, 2008 for an overview). Calculi developed in the general area of qualitative spatial and temporal representation and reasoning (QSTR) allow for meaningful processing of spatio-temporal information because they focus on categorical (discrete) changes or salient discontinuities (Egenhofer & Al-Taha, 1992; Galton, 2000) in the environment, which are thought to be relevant to an information processing system (both human and artificial). While qualitative calculi are naturally appealing and, on a general level, widely

^{*} This research is funded by the National Science Foundation (#0924534) and Deutsche Forschungsgemeinschaft (DFG) grant SFB/TR 8 Spatial Cognition.

acknowledged in both spatial and cognitive sciences (e.g. Kuhn, 2007; Lakoff & Johnson, 1980), there is comparatively little behavioral assessment of the cognitive adequacy of these calculi (see Klippel, Li, Yang, Hardisty, & Xu, in press and Mark, 1999 for overviews). To the best of our knowledge, there are no studies involving more than two entities at the same time, an observation that forms the motivation for the work described in this paper.

The two most prominent qualitative spatial formalisms in GIScience are arguably the 9-Intersection model (Egenhofer & Franzosa, 1991) and RCC-8 (Randell, Cui, & Cohn, 1992). Although, the underlying formalization is different in each approach, both formalisms make the same eight basic distinctions for topological relations holding between two simple regions in the plane (see Figure 1). When we look at applications of these and other qualitative models, many of them already employ or could benefit from incorporating a suitable notion of similarity between spatial configurations of objects. In querying and retrieval scenarios based on qualitative information (Papadias & Delis, 1997), for instance, a model of relational similarity allows for providing a ranked set of solutions (instead of returning just one solution).

The common approach to measure the similarity between two qualitative relations from the same qualitative calculus is based on so-called conceptual neighborhood graphs (CNG) (Egenhofer & Al-Taha, 1992; Freksa, 1992). CNGs are based on a notion of continuous change on a qualitative level (Galton, 2000) and two relations R_1 and R_2 are said to be conceptual neighbors if it is possible for R_1 to hold over a tuple of objects at a certain point in time, and for R_2 to hold over the tuple at a later time, with no other (third) mutually exclusive relation holding in between (Cohn, 2008). A CNG has one node for each relation and an edge between two nodes if the corresponding relations are conceptual neighbors. In Figure 1, the edges show the CNG structure of RCC-8 and the 9-Intersection model.

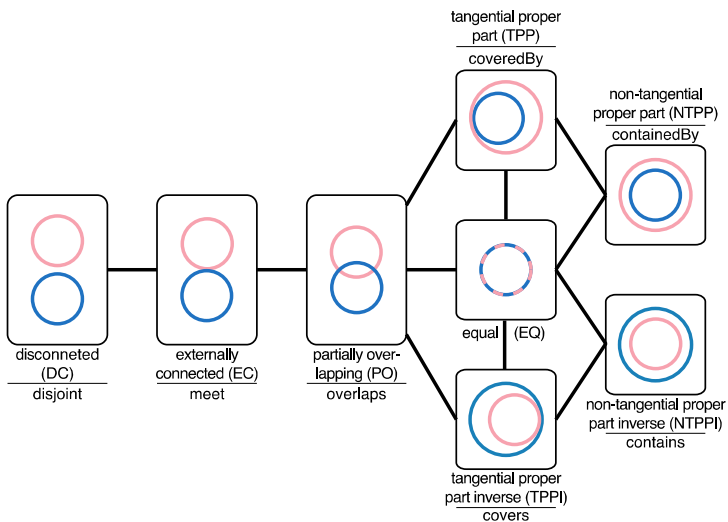


Fig. 1. Relations of RCC-8 and the 9-Intersection calculus arranged in accordance with their conceptual neighborhood graph (indicated by the edges)

Traditionally, the dissimilarity (or distance) $d(R_1, R_2)$ between two relations has been measured by assuming uniform weights for the edges in the CNG and counting the number of elementary changes or steps along the shortest connecting path in the CNG (Bruns & Egenhofer, 1996; Schwering, 2007). The dissimilarity of RCC-8 relations DC and PO, for instance, is 2 while it is 4 for DC and NTPP (see Figure 1). However, this simplistic approach has been challenged: On the one hand, researchers have developed alternative approaches using different weighting schemes, mainly based on intuition and introspection such as in the work by Li and Fonseca (2006): a weight of 3 is assigned to the edge between DC and EC, a weight of 2 for EC and PO, and a weight of 1 for TPP and NTPP. Only a few empirical investigations on the appropriateness of qualitative calculi using, for instance, grouping experiments with visual stimuli (Mark & Egenhofer, 1994) have been undertaken with the goal of painting a clearer picture of human relational similarity assessments and its relation to qualitative spatial formalisms. Related to this is the question whether the relational equivalence classes introduced by a qualitative calculus make the relevant distinctions to begin with or whether, for instance, coarser models such as RCC-5 or the coarse version of the 9-Intersection model (Knauff, Rauh, & Renz, 1997) should be preferred.

While progress has been made over the last years in evaluating the appropriateness of qualitative calculi and grounding similarity weighting of the respective relations in empirical data, we are facing a lack of similar work with respect to the problem of defining suitable similarity measures for complex spatial scenes. *Complex* is defined here as spatial configurations involving more than two objects. This fact is astonishing as such measures are urgently needed for application areas such as similarity-based querying and retrieval. Existing computational approaches (Bruns & Egenhofer, 1996; Dylla & Wallgrün, 2007; Papadias & Delis, 1997) compute similarities between qualitative equivalence classes (QECs) defined by the $m = n(n - 1)/2$ qualitative relations holding between n spatial entities by aggregating, in particular summing up, elementary neighborhood distances over corresponding relations, for example:

$$D(QEC_1, QEC_2) = \sum_{i=1}^m d(R_i^{[1]}, R_i^{[2]})$$

where $R_i^{[1]}$ and $R_i^{[2]}$ stand for the i th relation from QEC_1 and QEC_2 , respectively. With eight base relations in RCC-8, there exist 512 possible equivalence classes for three entities; but, only 193 of these are consistent QECs in the sense that they can be satisfied by actual triples of simple regions in the plane. Figure 2 shows approximately 15% of these 193 QECs depicted by an exemplary configuration of three ellipses with the respective qualitative relations listed on the side. The QECs for n entities can be connected to form a conceptual neighborhood graph (called CCNG for complex conceptual neighborhood graph) in the same way as the CNG for individual relations. The edges in the depicted CCNG connect those QECs in which exactly one of the relations has changed to a conceptual neighbor (e.g., EC to PO). The connected pairs of QECs are exactly those for which the dissimilarity

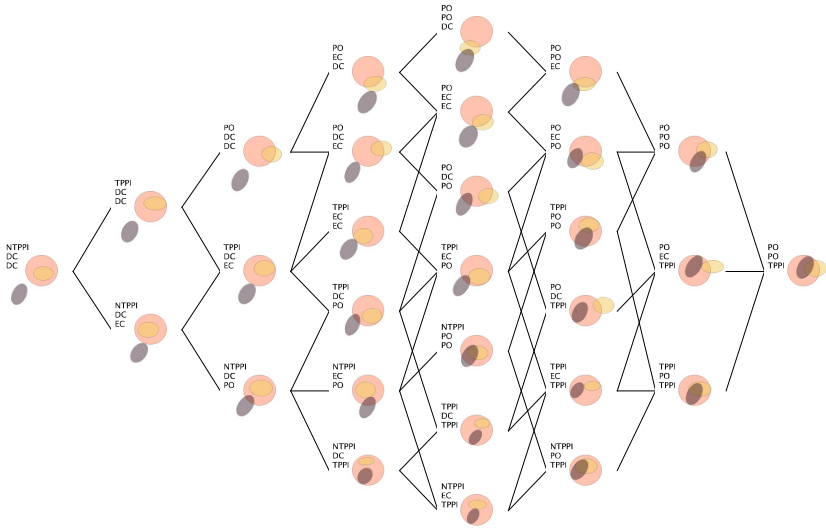


Fig. 2. Part of the RCC-8 / 9-Intersection complex neighborhood graph for three objects. Edges connect those QECs that have an aggregated distance of $D = 1$.

$D(QEC_1, QEC_2)$ is 1. This, however, raises many important questions with regard to a suitable choice for the involved aggregation operators as well as the appropriateness of the overall approach. An empirical basis to answer these questions, which can be expected to improve current implementations, is still largely missing.

The research described in this paper aims at remedying this situation by developing the empirical and methodological basis for evaluating and improving qualitative approaches for relational similarity assessments in complex scenes involving more than two objects. We report on three grouping experiments in which participants were given icons showing different configurations of three simple objects (Section 2). In our analysis (Section 3), we employ different clustering and cluster validation approaches to compare human similarity assessment (as an expression of cognitive conceptualizations) to the qualitative equivalence classes induced by topological calculi and evaluate the adequateness of the approach using $D(QEC_1, QEC_2)$ as defined above as a model of similarity.

2 Experiments

This section details three category construction (grouping) experiments that we conducted to shed light on human conceptualizations of spatial scenes with three entities. For the purpose of this paper, we define a spatial scene as a configuration of three spatially extended objects visually represented in a map-like format. Specifically, we used three elliptical entities such that each scene can be characterized using three topological relations (see Figure 2). Each of the three experiments was identical except for the semantic domain information that was associated with the scenes. The semantic domains we chose for this experiment in addition to a purely

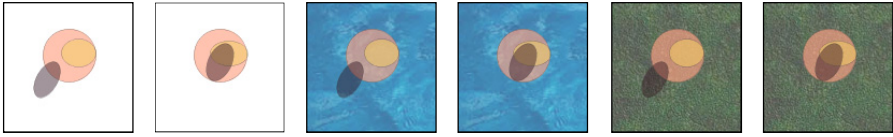


Fig. 3. Two exemplary icons for each of the domains: geometric, ocean, and forest

geometric one were: an ocean scenario with a blue water background and a forest scenario with a greenish background. The ellipses themselves were introduced as areas demarcating habitats of species. Figure 3 shows two instances of icons for each of the three domains (geometric, ocean, forest).

Design and Materials. The visual stimuli used in the experiments consist of three sets of 116 icons each, one set for each of the three semantic domains (see Figure 3). In our experiments, we consider the 29 QECs shown in the partial CCNG from Figure 2. These are all QECs that can be considered as being 'between' the QEC NTPPI-DC-DC on the left and the QEC PO-PO-TPPI on the right. For each of these 29 QECs we created four instances that were topologically identical but varied geometrically; hence, $4 \times 29 = 116$ icons. Each icon was 120x120 pixels in size. The geometric layout was randomized in the following way: We started with the geometric configurations representing the respective QEC in Figure 2 and randomized the following parameters of the two smaller ellipses: semi-major radius, semi-minor radius, x and y coordinate of center, and rotation angle. This was done using a uniform probability distribution over the interval $[p - \delta, p + \delta]$ where p is the value of the parameter in the prototype and delta is an individually chosen threshold value. The threshold values used were 10 pixels for the coordinates, 4 pixels for both radii, and 5 degree for the rotation angle. Because the random variation may change the qualitative relations holding between the objects, this step was followed by a brute force search within the parameter space for a set of parameters closest to the randomly generated parameter set and satisfying the qualitative relations given in the respective QEC. Possible parameter sets were constrained by the fact that some topological relations (e.g., NTPPI) are only possible for certain size relations between the involved entities.

We originally generated 10 geometric instances for each QEC and generated icon sets by drawing the ellipses on different backgrounds (white background for the purely geometric domain, and textured backgrounds for the other two domains) using transparency and a color scheme that would work well with all three different backgrounds. We then manually selected the first four icon instances for each QEC that were visually clear in the sense that the qualitative relations were deemed to be recognizable. It turned out that for three QECs additional instances had to be generated to get four clearly recognizable scenes.

Participants. Each experiment had 22 participants, Penn State students who received course credit for their participation. The female-to-male ratios were 11/11 for the

geometry condition, 8/14 for forest, and 11/11 for ocean (one participant had to be excluded because of providing bizarre information in the linguistic descriptions, see Procedure). The average age was 22.75, 20.05 and 21.05, respectively.

Procedure. The experiments were designed as group experiments and took place in a GIS lab. Up to 16 participants were able to take part in the experiments at the same time with workplaces separated by view blocks. Computers were Dell workstations with 24" widescreen LCD displays. The experiment was administered through our custom made software CatScan (Klippel, Li, Hardisty, & Weaver, 2010). Participants only grouped one of the three scenarios and were explicitly introduced to the semantics of the scenario that they were supposed to imagine. To ensure that they understood the task and semantics of the scenario, they had to enter keywords (e.g., forest, habitat) into the interface before they could start the experiment. Keywords were checked for their correctness. They also were given an unrelated category construction task (Medin, Wattenmaker, & Hampson, 1987) to acquaint themselves with the general idea of category construction and the interface. Participants then performed the category construction task on the stimuli. All 116 icons were initially presented on the left side of the screen with no groups on the right side. Participants were required to create all groups (as many as they thought appropriate) themselves. CatScan allows for icons to be moved around (into, out off, and between groups) by a simple mouse drag and drop procedure. After sorting all icons into group(s), participants were again shown the groups they had created and asked to provide a short linguistic label (max. 5 words) and a more detailed description of their grouping rationale.

3 Results

The data we collected in the three experiments comprised information about the categories each participant created in the form of binary matrices ranging over the icon sets containing a '1' if the respective icons were put into the same group and a '0', otherwise. These matrices form the basis for the analyses conducted and described in this section. In addition, the linguistic descriptions were collected in spreadsheets.

Our analysis and evaluation described in this section addresses the question of the influence of domain semantics as well as a detailed analysis of the category construction behavior of participants. The latter can be taken as a basis for evaluating existing approaches on defining similarities (semantics) of spatial scenes.

3.1 Comparison of Raw Similarities

To derive overall raw similarities for each of the three experiments, we combine the binary matrices from individual participants into a single overall similarity matrix (OSM) by summing up corresponding matrix cells. As a result, we get a matrix with

values from 0 for pairs of icons that were never put into the same group (and, hence, are rated as maximally dissimilar) to N (= number of participants; here: 22) for pairs that were put into the same group by all participants, considered to be maximally similar. Figure 4 illustrates the resulting OSMs in form of heat maps using colors from white (corresponding to 0) to red (corresponding to N). The entries are alphabetically ordered such that all 4 icons belonging to the same QEC correspond to a group of neighbored rows and columns in the matrices.

The heat maps allow for a first visual inspection of the grouping behavior. The red 4×4 squares along the diagonals of all three matrices are a clear indication that icons belonging to the same QEC are rated as being very similar, that is, they are (almost) always placed together into the same group. To back up this observation by numbers, we computed the sum over all entries for each block (QEC), took the average over all QECs, and normalized the result to be within $[0,1]$. The results show an average of 0.95 with 0.03 standard deviation for the purely geometric domain, 0.93 for ocean (standard deviation 0.04), and 0.94 for forest (standard deviation 0.04). This can be interpreted as evidence that topological equivalence classes potentially offer an explanation of how humans conceptualize spatial scenes. Additionally, however, there are several other areas in the OSMs with high similarities. This is a first indication that the 29 topologically defined equivalence classes form coarser conceptual groups.

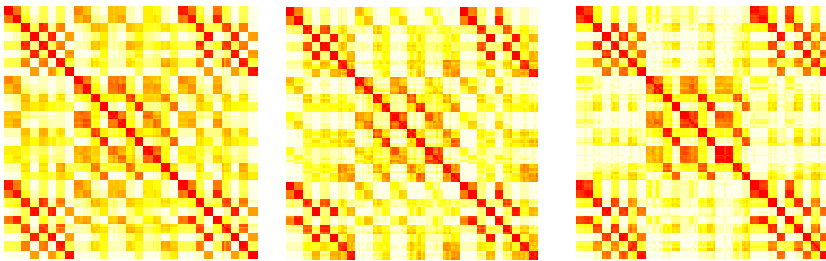


Fig. 4. Heat maps showing raw similarities (red = maximally similar, white = maximally dissimilar) for geometry (left), ocean (middle) and forest (right)

Further comparison of the heat maps in Figure 4 shows that overall the three patterns are very similar. To make, however, the differences more explicit, we computed difference-matrices for each pair of OSMs using the operation $\text{abs}(\text{OSM}_1 - \text{OSM}_2)$ for each cell. The resulting matrices are shown in Figure 5 emphasizing where differences do exist. Computing the average differences over all entries (except the diagonals which have to be zero) and normalizing them to $[0,1]$, we get the following results: 0.08 for geometry-ocean, 0.08 for geo-forest, and 0.09 for forest-ocean. This means that the difference in similarity assessment averaged over all pairs of icons is less than 9% between the domains. This is a very low number given that within each domain individual differences exist, too.

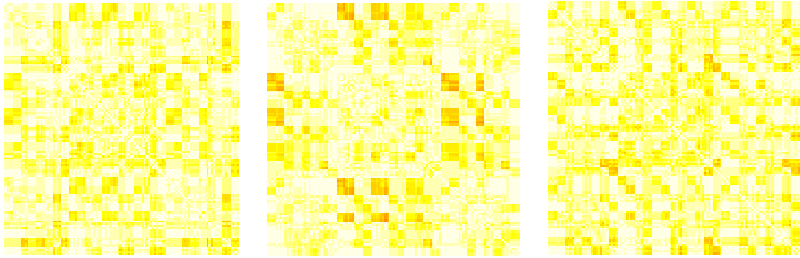


Fig. 5. Heat maps showing the differences between OSM matrices for geometry-ocean, geometry-forest, and forest-ocean (white = 0 difference; maximal difference would be red but does not occur)

3.2 Clustering

We followed widely accepted procedures on cluster analysis and cluster validation, that is, for each scenario we performed three different types of cluster analysis (Ward's methods, average linkage, complete linkage) and compared the clustering structure (Kos & Psenicka, 2000). The resulting clusterings can be visualized as tree structures called dendrograms in which the leaf nodes represent the individual icons (instances of a QEC). Figure 6 shows a small part of such a dendrogram. We found large similarities between the different scenarios but also dissimilarities especially comparing different methods. The reasons for these differences seem to be largely unrelated to the semantics of a particular domain but are the results of a more complex decision space: We have 29 QECs with four instances for each QEC. Looking into how hierarchical cluster algorithms operate, we find that initial similarities/dissimilarities can lead to different clustering structures reinforced by the recalculation of similarities after each clustering step; these differences are not reflective of high overall similarities (as Figure 5 shows that all scenarios are very similar). Given space constraints, it is not possible to discuss all nine cluster analyses (three for each experiment/domain) in detail. However, to harvest what cluster analysis reveals about the similarities / categories of complex spatial scenes, we developed a method that we consider highly valuable for researchers evaluating results of clustering methods. With this method that we term *greatest common divisor* algorithm, we are able to identify the most fundamental category construction aspects (similarities) across all nine cluster analyses (three scenarios with three cluster analyses each).

An important prerequisite is that QECs are very strong predictors for category construction, that is, instances of a QEC are not separated in any of the clustering methods (compare Section 3.1). Either all icons of the same QEC are combined into a single cluster before the resulting cluster is combined with icons from a different QEC; or, icons from two or, in a few cases, three conceptually neighbored QECs are joined in a merged way forming a single cluster. This is another indication that topological relations and conceptual neighborhood graphs capture important factors of the cognitive conceptualization of spatial scenes.

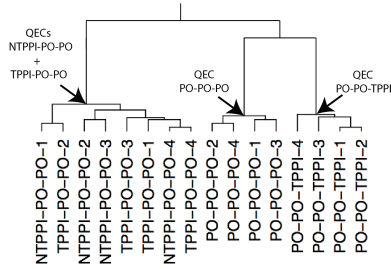


Fig. 6. Part of a dendrogram from a clustering method. The leaf nodes represent the icons (instances of QECs) which are combined to form larger groups on higher levels.

```

1: procedure GREATESTCOMMONDIVISOR( $\mathcal{D}$ )


---


2:   Input:
3:    $\mathcal{D}$  set of dendrograms  $D_i$  with leaf nodes annotated with  $\{s\}$  where  $s$  is the respective icon name
   ;; initialization


---


4:   do
5:     find a node  $N$  on level  $\text{depth}(D_i) - 1$  in a  $D_i \in \mathcal{D}$  and with either
       not all child nodes of  $N$  be labeled with the union of all icons of a set of QECs
       or the label of a child node of  $N$  is a strict subset of the label of a leaf node in another  $D_j$ 
6:     call MERGE( $D_i, N$ )
7:   until no such node can be found
   ;; clustering
8:   do
9:     find one  $N_i$  for each  $D_i \in \mathcal{D}$ , all on level  $\text{depth}(D_i) - 1$  and with identically labeled child nodes
10:    call MERGE( $D_i, N_i$ ) for all  $i$ 
11:  until no such set of nodes can be found
12: end procedure


---


1: procedure MERGE( $D, N$ )


---


2:   Input:
3:    $D$  tree
4:    $N$  node in  $D$  at level  $\text{depth}(D) - 1$ 


---


5:   annotate  $N$  with the union of annotations of its child nodes
6:   remove all child nodes of  $N$ 
7: end procedure

```

Algorithm 1. Algorithm to derive largest common clusters.

Now that we know that individual QECs are potential category predictors, we seek to find QECs most similar to each other. To this end, we continued the bottom up analysis of consistent clustering results across all three scenarios and all clustering methods using the greatest common divisor algorithm shown in Alg. 1. This algorithm aims at determining the largest groups of QECs for which the order of combination is identical over all three experiments and all three clustering methods. It consists of two phases: the initialization and the main loop (clustering).

In the initialization phase, the algorithm merges leaf nodes starting with the individual icons, until we have nine tree structures with identical leaves in terms of associated icons, and each leaf represents all icons from one or more QECs. In Figure 6, for instance, we end up with the new leaf nodes marked by the arrows with the left one representing two QECs and the other two representing individual QECs.

In the main loop, the algorithm combines leaf nodes if they exist in all trees and are connected to the same parent node, meaning that clusters are merged in the same local order along rising branches in all dendrograms. The result of this procedure applied to our nine dendrograms is shown in Fig. 7 (colored lines surrounding QECs). Several larger groups are identified using this algorithm that are plausible from the perspective of applying a coarser calculus such as RCC-5 (we will come back to this discussion in the conclusion/discussion section). However, we also find that several parts of the CCNG are split up to form clusters with only one or two QECs. To address this issue, we modified Alg. 1 to use a less restrictive criterion in line 8: We allowed merging when the respective groupings were combined in at least six out of nine cluster analyses. Using this modification, we were able to reveal some essential aspects of the grouping behavior of participants across all three experiments.

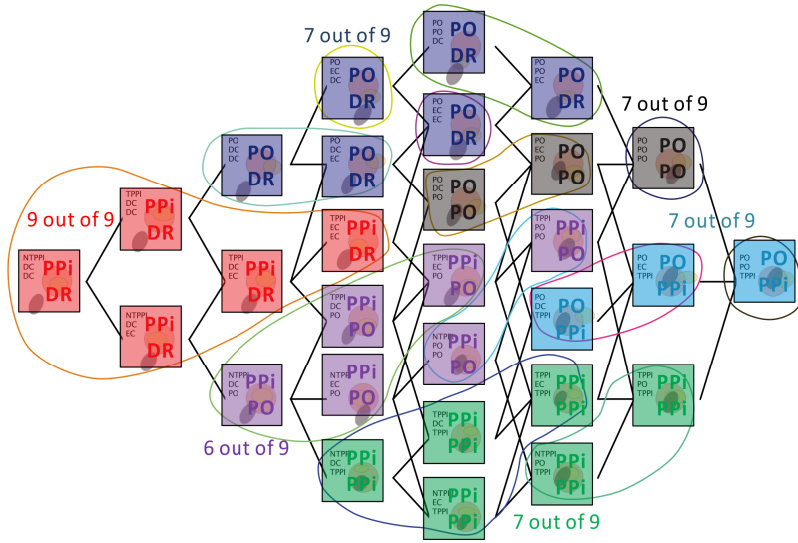


Fig. 7. Greatest common divisor clusters identified by Algorithm 1. Colored lines indicate results using hard constraints, that is, clusters found in all 9 dendrograms. Colored boxed indicate relaxed constraints (minimum 6 out of 9 dendrograms). Colored letters (e.g., DR) indicate RCC-5 relations (not for relations between small ellipses).

To further visualize the results, we annotated Figure 7 in the following way: 1) we used a color coding to identify the six resulting cluster; 2) each scene has been characterized based on a coarser level of granularity (bold letters) using a) only the relation between the large circle and each of the smaller ellipses (but not the relations between the two smaller ellipses); b) instead of using RCC-8, we used RCC-5 which combines DC and EC into DR (discrete) and NTPP and TPP to PP (proper part).

With these annotations and reanalysis we were able to reveal an astonishingly clear picture of the grouping behavior of participants: The conceptualization process of participants centered on two aspects: First, relations were simplified according to

RCC-5, second, the important distinctions were made looking into the relationship between each small ellipse with the larger circle individually while the relation between the smaller ellipses was largely ignored. This strategy was not scenario specific as in most cases seven (or more) cluster methods support this interpretation.

3.3 Linguistic Analysis

We performed a word-count analysis on the linguistic descriptions collected using AntConc, a corpus analysis toolkit (Anthony, 2011). Before the analysis, we excluded spatially irrelevant words such as colors (e.g., red, black, yellow), pronouns (e.g. this, these, which), words referring to the entities (e.g., ellipse, oval, habitat), and other common English words (e.g., is, but, and). In addition, we combined the frequencies of the same word in different tenses (e.g., overlap, overlaps, overlapping, and overlapped), and also synonyms (e.g., completely, fully, totally, and entirely). The final results are shown in Table 1.

Table 1. Top 10 frequently mentioned words from participants' linguistic description

Rank	Geometry		Forest		Ocean	
	Word	Frequency	Word	Frequency	Word	Frequency
1	in	130	overlap	116	overlap	102
2	overlap	97	in	114	in	91
3	partially	90	completely	96	both/two	55
4	both/two	89	partially	80	inside	54
5	completely	89	both/two	76	completely	49
6	inside	86	inside	73	within	49
7	out	69	with	51	outside	46
8	touching	58	not	47	all	45
9	outside	53	outside	47	not	44
10	not	41	all	30	with	37

First, it is noteworthy that the words “in”, “inside”, “out”, and “outside” are most frequently mentioned across the three domains. This suggests that the non-overlapping relation (DC and EC) are distinguished from overlapping relations (TPPI and NTPPI). Second, the only word referring to connecting relations (EC and TPPI) is “touching” (ranked 9th in geometry scene), which may indicate that the connecting relation is more relevant in the geometric domain (compared to forest and ocean). Third, “both” and “two” are frequently used by participants across all semantic domains. By additionally looking into the original descriptions, we found that, in most cases, these two words are used to describe the relations of the two smaller entities to the larger entity in each scene. The abovementioned findings support the conclusions we drew from the cluster analysis, i.e., participants' overall grouping rationale relies on RCC-5 and the relation between two smaller entities is often ignored.

4 Discussion and Conclusions

Constructing categories is arguably one of the most fundamental abilities that humans possess. Paralleling this aspect, the disciplines of the spatial sciences focus strongly on conceptualization and categorization to structure spatial as well as temporal information, often using ontological frameworks (Bateman, Hois, Ross, & Tenbrink, 2010). In the spatial sciences and related branches of artificial intelligence, qualitative spatio-temporal representation and reasoning formalisms play a prominent role in connecting human category construction with formal approaches to advance processes at the human-machine interface (representation, reasoning, retrieval).

The research reported in this paper closes an important gap: While approaches on simple configurations exist, no data is available on more complex scenarios, here: relations between three entities. Explorations into more complex and real world scenarios are important: First, because discontinuities identified by qualitative calculi focusing on two relations may behave differently in complex scenes with more relations (e.g., similarities/dissimilarities may or may not be adding up directly); and second, because it is not clear whether and how domain semantics influence the conceptualization of static spatial relations (see Coventry & Garrod, 2004 for a general discussion and Klippel, accepted for dynamic processes).

The results reported here can be summarized as follows: 1) Topological equivalence is a strong grouping criterion / category predictor. This is prominently demonstrated by the analysis of the grouping behavior of instances within QECs that are almost always placed together into the same groups. 2) Overall, the similarities between all three scenarios are highly indicating that—in this static case—the semantics of individual domains may not play a substantial role on the construction of categories of spatial relations, at least not for the domains chosen here. This analysis is reinforced by the linguistic descriptions provided by participants. They reveal that participants placed a strong focus on purely spatial aspects rather than incorporating domain specific language (other than referring to ellipses by using their color). 3) As the decision space gets more complex in CCNGs, there is more variation across different experiments and classic clustering methods are not necessarily well suited to distinguish commonalities from differences. To address this issue, we designed an algorithm that revealed the most fundamental coarse categories constructed by participants by comparing (here) nine different cluster analyses (three for each experiment/domain). We were able to demonstrate, clearly, two factors that explain the category construction behavior of participants: RCC-5 works well as a predictor of category membership taking additionally into account that the largest entity was used as a reference. As a result, the relations between the two smaller ellipses only played a subordinate role. We found the clarity of these results quite surprising. 4) Within all experiments and all cluster analyses (the original nine, not the aggregated one), we did never find a violation of category membership induced by the CCNG. In other words, all members of groups identified in the nine cluster analyses are always neighbors in the CCNG. This is probably one of the most promising results as it adds to the validity of using CCNGs for similarity assessments and category prediction.

These results support existing theories on conceptualization and category construction for spatial and non-spatial information. It has been a long debate how humans deal with complexity (Heil & Jansen-Osmann, 2008). Across different

disciplines it is generally assumed (and experimentally confirmed) that humans will reduce complexity and lower the individual pieces of information that they have to deal with (Cowan, 2001). In the end, this is what categorization is all about. What is less clear is which mechanisms they use and how to formally describe them such that they may be used in artificial systems, too. Two approaches are worth considering: a) participants could try to holistically assess the similarity of the scenes we presented them with; b) participants single out a particular dimension along which they construct categories (Pothos & Close, 2008). While both approaches are mutually exclusively discussed in the literature, our results seem to indicate that participants used a combination of both strategies. On the one hand, they singled out aspects (dimension in a looser interpretation) that they were able to use as anchors to categorize the scenes, specifically, a reduction of three relations to two by ignoring the relations between the smaller ellipses. On the other hand, they holistically simplified the scenes by ignoring RCC-8 and adopting a coarser perspective that can be captured by RCC-5.

Based on the promising results we will pursue this line of research to assess spatial similarity on different levels of scene complexity to advance approaches to formalize spatial semantics. We will perform additional experiments with, for example, varying domains and relaxation of the spatial constraints which we applied in the current experiments (e.g., to include additional aspects of spatial knowledge). One critical topic will be to investigate how the similarity measures derived from behavioral data can be transformed best into weights in (complex) conceptual neighborhood graphs.

References

- Anthony, L.: AntConc (version 3.2.2). Waseda University, Tokyo (2011), <http://www.antlab.sci.waseda.ac.jp/>
- Bateman, J.A., Hois, J., Ross, R., Tenbrink, T.: A linguistic ontology of space for natural language processing. *Artificial Intelligence* 174(14), 1027–1071 (2010)
- Bruns, H.T., Egenhofer, M.J.: Similarity of spatial scenes. In: Kraak, M.J., Molenaar, M. (eds.) *Seventh International Symposium on Spatial Data Handling (SDH 1996)*, Delft, The Netherlands, pp. 173–184 (1996)
- Cohn, A.G.: Conceptual neighborhood. In: Shekhar, S., Xiong, H. (eds.) *Encyclopedia of GIS*, p. 123. Springer, Boston (2008)
- Cohn, A.G., Renz, J.: Qualitative spatial representation and reasoning. In: van Harmelen, F., Lifschitz, V., Porter, B. (eds.) *Foundations of Artificial Intelligence. Handbook of Knowledge Representation*, 1st edn., pp. 551–596. Elsevier (2008)
- Coventry, K.R., Garrod, S.: Towards a classification of extra-geometric influences on the comprehension of spatial prepositions. In: Carlson, L.A., van der Zee, E. (eds.) *Functional Features in Language and Space*. Oxford University Press (2004)
- Cowan, N.: The magical number 4 in short term memory. A reconsideration of storage capacity. *Behavioral and Brain Sciences* 24, 87–186 (2001)
- Dylla, F., Wallgrün, J.O.: Qualitative spatial reasoning with conceptual neighborhoods for agent control. *Journal of Intelligent and Robotic Systems* 48(1), 55–78 (2007)
- Egenhofer, M.J., Al-Taha, K.K.: Reasoning about Gradual Changes of Topological Relationships. In: Frank, A.U., Formentini, U., Campari, I. (eds.) *GIS 1992. LNCS*, vol. 639, pp. 196–219. Springer, Heidelberg (1992)

- Egenhofer, M.J., Franzosa, R.D.: Point-set topological spatial relations. *International Journal of Geographical Information Systems* 5(2), 161–174 (1991)
- Freksa, C.: Temporal reasoning based on semi-intervals. *Artificial Intelligence* 54(1), 199–227 (1992)
- Galton, A.: *Qualitative spatial change*. Spatial information systems. Oxford Univ. Press, Oxford (2000)
- Heil, M., Jansen-Osmann, P.: Sex differences in mental rotation with polygons of different complexity: Do men utilize holistic processes whereas women prefer piecemeal ones? *The Quarterly Journal of Experimental Psychology* 61(5) (2008)
- Klippel, A.: Spatial information theory meets spatial thinking - Is topology the Rosetta Stone of spatio-temporal cognition? *Annals of the Association of American Geographers* (67 manuscript pages) (accepted)
- Klippel, A., Li, R., Hardisty, F., Weaver, C.: Cognitive Invariants of Geographic Event Conceptualization: What Matters and What Refines? In: Fabrikant, S.I., Reichenbacher, T., van Kreveld, M., Schlieder, C. (eds.) *GIScience 2010*. LNCS, vol. 6292, pp. 130–144. Springer, Heidelberg (2010)
- Klippel, A., Li, R., Yang, J., Hardisty, F., Xu, S.: The Egenhofer-Cohn Hypothesis: Or, Topological Relativity? In: Raubal, M., Frank, A.U., Mark, D.M. (eds.) *Cognitive and Linguistic Aspects of Geographic Space - New Perspectives on Geographic Information Research* (in press)
- Knauff, M., Rauh, R., Renz, J.: A Cognitive Assessment of Topological Spatial Relations: Results from an Empirical Investigation. In: Frank, A.U. (ed.) *COSIT 1997*. LNCS, vol. 1329, pp. 193–206. Springer, Heidelberg (1997)
- Kos, A.J., Psenicka, C.: Measuring cluster similarity across methods. *Psychological Reports* 86, 858–862 (2000)
- Kuhn, W.: An Image-Schematic Account of Spatial Categories. In: Winter, S., Duckham, M., Kulik, L., Kuipers, B. (eds.) *COSIT 2007*. LNCS, vol. 4736, pp. 152–168. Springer, Heidelberg (2007)
- Lakoff, G., Johnson, M.: *Metaphors we live by*. University of Chicago Press, Chicago (1980)
- Li, B., Fonseca, F.: TDD: A comprehensive model for qualitative spatial similarity assessment. *Spatial Cognition and Computation* 6(1), 31–62 (2006)
- Mark, D.M.: Spatial representation: A cognitive view. In: Maguire, D.J., Goodchild, M.F., Rhind, D.W., Longley, P.A. (eds.) *Geographical Information Systems: Principles and Applications*, 2nd edn., vol. 1, pp. 81–89 (1999)
- Mark, D.M., Egenhofer, M.J.: Calibrating the meanings of spatial predi-cates from natural language: Line-region relations. In: Waugh, T.C., Healey, R.G. (eds.) *Advances in GIS Research*, pp. 538–553 (1994)
- Medin, D.L., Wattenmaker, W.D., Hampson, S.E.: Family resemblance, conceptual cohesiveness, and category construction. *Cognitive Psychology* 19(2) (1987)
- Papadias, D., Delis, V.: Relation-based similarity. In: *Proceedings of the 5th ACM Workshop on GIS, Las Vegas*, pp. 1–4. ACM (1997)
- Pothos, E.M., Close, J.: One or two dimensions in spontaneous classifica-tion: A simplicity approach. *Cognition* (2), 581–602 (2008)
- Randell, D.A., Cui, Z., Cohn, A.G.: A spatial logic based on regions and connections. In: Nebel, B., Rich, C., Swartout, W.R. (eds.) *Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning*, pp. 165–176. Morgan Kaufmann, San Francisco (1992)
- Schwering, A.: Semantic similarity of natural language spatial relations. In: *Conference on Artificial Intelligence and Simulation of Behaviour: Artificial and Ambient Intelligence. Symposium: Spatial Reasoning and Communication* (2007)

A Qualitative Bigraph Model for Indoor Space

Lisa A. Walton and Michael Worboys

Department of Spatial Information Science and Engineering
University of Maine, Orono, ME USA
{lisa.walton, worboys}@spatial.maine.edu

Abstract. Formal models of indoor space for reasoning about navigation tasks should capture key static and dynamic properties and relationships between agents and indoor spaces. This paper presents a method for formally representing indoor environments, key *indoor events* that occur in them, and their effects on the topological properties and relationships between indoor spaces and mobile entities. Based on Milner’s bigraphical models, our *indoor bigraphs* provide formal algebraic specifications that independently represent agent and place locality (e.g., building hierarchies) and connectivity (e.g., path based navigation graphs). We illustrate how the model supports the description of scenes and narratives with incomplete information, and provide a set of reaction rules dictating legal system transformations to support goal-directed navigation. Given a starting scene and a particular navigation task we can determine potential sequences of events satisfying a goal (e.g., if a building fire occurs, what actions can an agent take to reach an exit?).

Keywords: Bigraphs, Bigraphical Reactive Systems, Indoor Bigraphs, Indoor Events, Indoor Space, Indoor Navigation.

1 Introduction

Spatial information systems supporting indoor navigation require models of built space that transcend traditional 3D CAD or building information models (BIMs). Although BIMs support the representation of building elements in terms of their 3D geometric and non-geometric (functional) attributes and relationships [1], they do not typically provide support for modeling navigation tasks. Outdoor navigation systems usually incorporate 2.5D models of large scale geographic environments consisting of physically bounded 2D regions (e.g., building footprints and cities) with an extra half dimension attribute (e.g., elevation) which are overlaid by road networks. Mobile objects of interest (e.g., pedestrians or cars) are represented by points (or single icons) that move through the network or inside the flat regions. Outdoor systems, however, cannot typically support indoor navigation, which must take topological configurations such as building hierarchies into account. In addition, locality, often defined with absolute coordinates in outdoor environments, is more likely to be described in relative terms for both physical and functional spaces in indoor environments (e.g., John’s office). Moreover, traditional approaches to modeling built

spaces typically require quantitative (geometric) building and navigation data. Existing spatial models supporting indoor navigation often incorporate graph-based representations of architectural space, and some consider cognitive models of agent navigation behavior in indoor environments [2-5]. However, many of these do not formally capture connections between the network model for navigation and other, non-spatial relations that exist between indoor spaces and mobile objects or agents. Our approach constructs a qualitative model of indoor environments based on Milner's bigraphs [6], which provide a formal method for independently specifying connectivity and locality between nodes (places of interest). Milner's tight coupling of place and connectivity graphs combined with formal methods for modifying and composing bigraphs provide a novel and useful approach for representing and reasoning about indoor navigation.

This paper presents a qualitative framework for formally representing and reasoning about indoor environments to support indoor navigation tasks. It models agent actions and their effects on topological properties and relationships between indoor spaces and mobile objects and agents. Indoor bigraphs provide formal algebraic specifications of indoor environments that independently represent agent, object, and place locality (e.g., building hierarchies) and connectivity (e.g., path based navigation graphs). The framework is flexible enough to model and reason about indoor scenes with incomplete information. System configurations can be updated with more complete scene information or in response to an agent's dynamic behavior as they carry out goal-directed indoor navigation tasks. In the following sections we define bigraphs and show how to build indoor bigraphs using floor plans and contextual knowledge about indoor scenes. Finally, we describe modifying indoor bigraphs based on new contextual information or changes due to agent actions. The material presented here complements earlier work by the authors on cognitive representations of indoor space and spatial relations in bigraphs [7] using image schema such as CONTAINER and PATH, and section 5 includes examples of reaction rules (which model atomic agent actions) using image schema.

2 Bigraphs

Originally developed for the virtual world of communicating processes and information objects, bigraphs originate in process calculi for concurrent systems, especially the pi-calculus [8] and the calculus of mobile ambients [9] for modeling spatial configurations (e.g., networks with a dynamic topology). Ambients, represented as nodes in bigraphs, were originally defined as "bounded places where computation occurs" [10]. However, bigraphs nodes typically have a more general interpretation as bounded physical or virtual entities or regions that can contain or link to other entities and regions. In defining indoor bigraphs here we will not repeat Milner's formal definitions, instead we use his simpler visual descriptions that are tightly coupled with an underlying algebra that provides reaction rules for appropriate system transformations based on connection or location changes in spatial configurations.

A **bigraph** G is a pair of constituent independent graphs $G = \langle G^P, G^L \rangle$ sharing a common set of nodes where *place graph* G^P specifies containment relations and *link graph* G^L specifies connectivity relations. Fig. 1 shows a bigraph and its constituent place and link graphs with nodes $\{A,B,C\}$ and a single undirected hyperedge ABC.

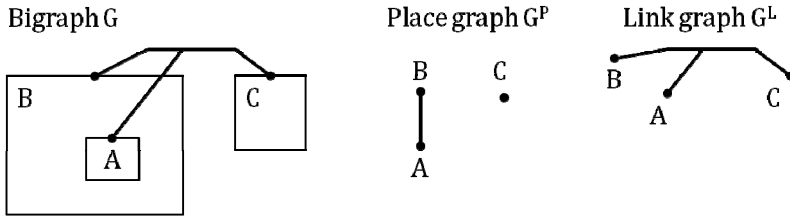


Fig. 1. Bigraph G with place graph (downward directed) and link graph (undirected)

Containment relations in bigraphs are visualized by letting nodes contain other nodes and connectivity relations by hyperedges joining two or more nodes. Bigraph visualizations need not maintain region shapes, relative node positions, or relative node sizes. Other spatial relationships between regions such as overlap, meet, or equals are not expressible as place relations in basic bigraphs.

Place graphs are forests of trees showing only containment relations between places (e.g., in Fig. 1 G^P has two trees). In general, a place is a *node*, a *root* (outer context) or a *site* (inner context). Here, contexts are not actual nodes (bounded regions or entities), but rather *open placings* for partially known scene information that can be filled in later via bigraph composition (see section 4). For example, in indoor bigraphs (see next section) a root might represent the unknown environment outside a building and a site a collection of unknown objects in a room. Staying with the simple example, if we knew B and C were set in the same (but unknown) outer environment, and that B and C had unknown content, we could model that as follows (Fig. 2):

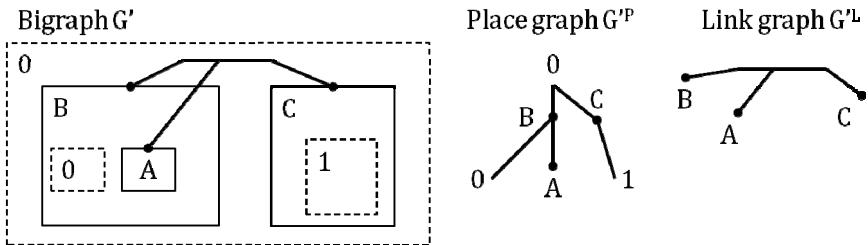


Fig. 2. Adding a root and sites to bigraph G

By convention [6], roots and sites are both labeled with integers beginning at 0. Note that when additional open placings are added the link graph remains unchanged.

Link graphs are hypergraphs, generalizations of graphs in which an edge may join any number of nodes. A *link* is a hyperedge connecting *nodes*, *inner names*, or *outer*

names, where names are *open linkings* that support additional connectivity. Open linkings, like open placings, support the addition of context via bigraph composition. The bigraph and link graphs above both contain a single undirected hyperedge ABC.

3 Indoor Bigraphs

Indoor bigraphs to support navigation tasks must represent place and accessibility relations between people, objects, and spaces based on environmental features (e.g., doors may be locked), and agent capabilities (e.g., agent may not have appropriate keys or an agent may not be able to use stairs). Indoor *bigraph nodes* represent spatio-temporally bounded entities including rooms, people, and mobile small-scale objects such as keys. Fig. 3 describes an initial indoor scene with a first floor plan and an agent in the reception area (RA) who has a key to room 102.

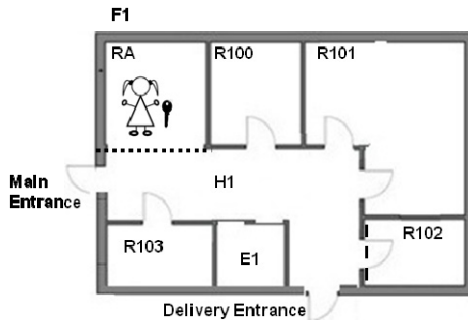


Fig. 3. First floor plan and agent and key initial locations

The bigraph for this scene will have ten nodes {F1, RA, R100, R101, R102, R103, E1, H1, A, K102}. The agent A, the key K102, and six spaces (the floor itself (F1), rooms 100-103, and the elevator E1) are completely physically bounded, and the hallway (H1) and reception area (RA) share a fiat boundary. Note that there are doors to the outside, and that the elevator presumably can access at least one other floor. Fig. 4 shows the **place graph** $F1^P$ for this scene as a tree with root 0 (outer context where the exits lead to), nodes, and one site 0 (unknown context in room 102).

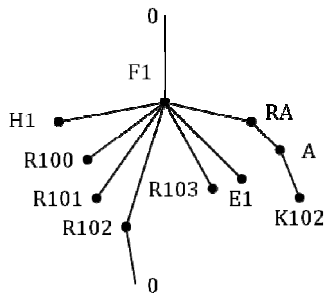


Fig. 4. Place graph $F1^P$ for the first floor scene

Although the elevator probably leads to a different place than the building exits, we defer representing that partial information to a later section when we have more building information. While there are many ways to represent a locked room scenario in a bigraph [7], here we have chosen to represent the agent-key relation as a place relation (i.e., the key is *in* the possession of the agent), and the key-lock relation as a link.

Indoor Accessibility Graphs

Most floor plans can be discretized to obtain an adjacency or accessibility graph [3]. For many indoor navigation tasks accessibility is sufficient to determine if an agent can navigate between places. Fig. 5 shows a discretization of the first floor scene.

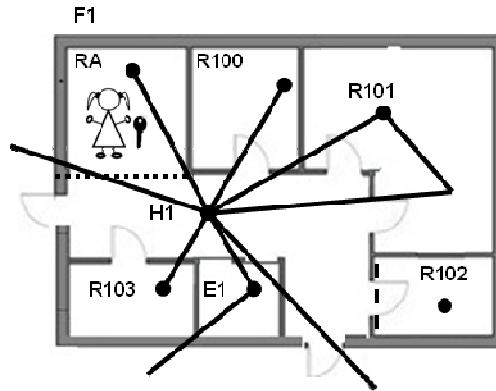


Fig. 5. Accessibility graph for the first floor

Rooms with two doors (e.g., R101) require two edges. Edges to unknown areas (the outdoors and other building floors) are open. R102 is locked (inaccessible) from the rest of the floor. Combining the accessibility graph and the single dotted edge key-lock graph yields the **link graph $F1^L$** in Fig. 6.

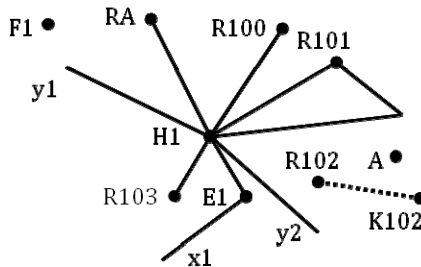


Fig. 6. Link Graph $F1^L$ for the first floor

The first floor (F1) and agent (A) nodes are in the link graph, but have no link relations. Open links to unknown areas are labeled with outer names y_1 , y_2 (to the outdoors) and inner name x_1 (to another floor). Combining the link and place graphs yields the bigraph in Fig. 7. Note that the elevator node has been moved to the top of the diagram for readability since topological configurations need not be preserved in bigraph representations. Node types are visualized using solid bordered squares to represent spaces, triangles for agents, and circles for mobile objects such as keys. Open placings (roots and sites) are represented as regions with dashed lines.

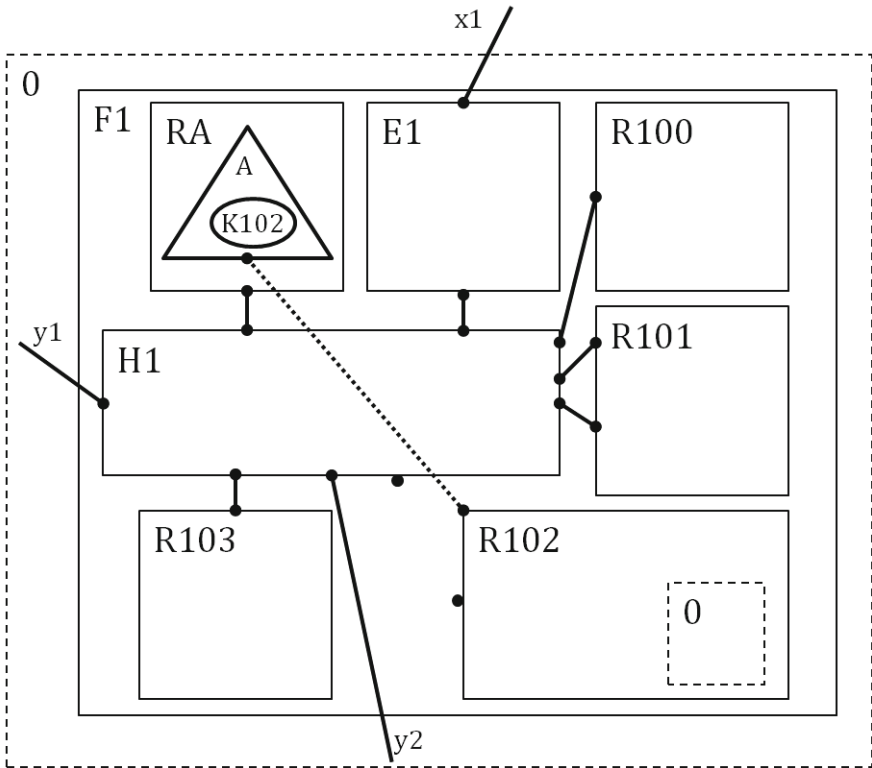


Fig. 7. Bigraph F1: $\langle \{0\}, \{x_1\} \rangle \rightarrow \langle \{0\}, \{y_1, y_2\} \rangle$

Each node has a fixed number of *ports* indicating the number of links (of any type) that are permitted, and any port can be connected to 0 or 1 edge. Here, the agent and first floor have no ports. Rooms 100, 103, and the RA each have 1 port, whereas room 101 with two exits, the elevator, and lockable room 102 each have 2 ports. The hallway has 9 ports. Edge types are visualized with solid lines for accessibility relations and dotted lines for key-lock relations. Indoor bigraph closed edges connect exactly 2 ports, and open linkings (inner and outer names) connect to 1 port.

Each bigraph has a mapping between *interfaces*, or minimal specifications of the portions of a particular bigraph that support additional openings for more containment

or linking information. Bigraph F1 above has site set $\{0\}$ and inner name set $\{x1\}$ that map to root set $\{0\}$ and outer name set $\{y1,y2\}$. Ports, including open ports such as those on the hallway and R102, are not included in the interface.

3.1 Indoor Bigraph Typology

For the domain of indoor navigation the typology of places (nodes) and edges is very important. So, for each specific indoor environment at a minimum the following bigraph node types must be defined:

- $A = \{a_1, \dots, a_i\}$ is a finite set of agents
- $P = \{p_1, \dots, p_j\}$ is a finite set of places an agent can be in
- $K = \{k_1, \dots, k_k\}$ is a finite set of keys where k_i unlocks place p_i

The set of bigraph nodes is $N = A \cup P \cup K$. Edge types are defined as follows:

- $AEdges = \{(p1,p2) \mid p1,p2 \in P\}$ is a finite set of edges representing accessibility relations in the link graph
- $KEEdges = \{(k,p) \mid k \in K, p \in P\}$ is a finite set of edges representing key-lock relations in the link graph
- $PEEdges = \{(p1,p2) \mid p1,p2 \in P\}$ is a finite set of edges representing containment relations in the place graph

The set of bigraph edges is $E = AEdges \cup KEEdges$. PEEdges in bigraph diagrams are represented as actual region containment not with an edge. Note that for indoor bigraphs all edges are between just two nodes, although the general bigraph model allows hyperedges. Typically, nodes sets will also be partitioned according to how many ports (possible link connections) each type of node can support.

The bigraph example above is a temporal snapshot of an incomplete indoor environment. While floor plans are usually static, agent locations and the lock state of doors (and hence access links) can change over time. In the following sections we first show how to add context via bigraph composition and next how to make changes to indoor environments (e.g., people moving around) by applying reaction rules to change system configurations. Bigraphs can be modified using:

- *Composition* to add missing context (e.g., create outdoor-indoor bigraphs or combine partial building plans)
- *Reaction rules* to change the system state (e.g., person unlocks a room)

4 Bigraph Integration

4.1 Bigraph Composition

Given two bigraphs with matching interfaces, composition is used to add additional context. For example, outdoor spaces containing buildings and road networks can be integrated with indoor spaces. Suppose building B1 is accessible from two parking lots which access the city road network. Fig.8 specifies an outdoor scene including a footprint for the building containing the first floor.

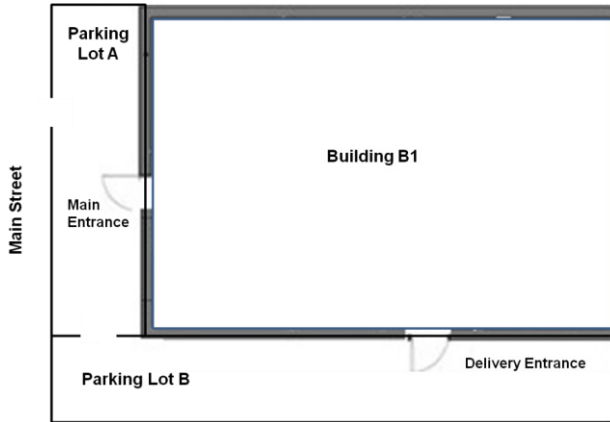


Fig. 8. Outdoor Map

Resolving Outer Names and Roots

Suppose that we know that the entrances from the parking lots lead to the 1st floor hallway, represented as open links (outer names) y_1 and y_2 in the 1st floor bigraph. In addition, we know there are two floors in the building, the first of which is inside root 0 in the original bigraph. Adding sites 0 and 1 (placeholders for the 1st and 2nd floors) we define a *host bigraph* H which can be composed with the original bigraph to provide outside context. Fig. 9 shows an outdoor bigraph H with appropriate interfaces.

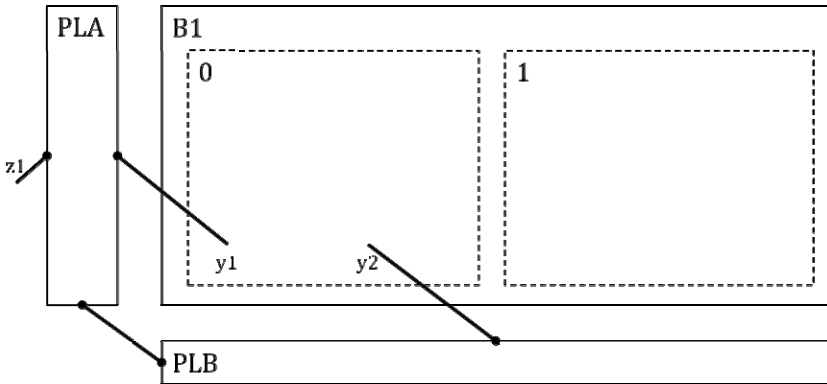


Fig. 9. Outdoor bigraph $H: \langle \{0,1\}, \{y_1, y_2\} \rangle \rightarrow \langle \{\}, \{z_1\} \rangle$

Fig. 10 shows the composition of the 1st floor bigraph F_1 and the outdoor bigraph H . It joins root 0 in F_1 with site 0 in H causing both to disappear, and joins the open links y_1 and y_2 replacing them with two closed edges between the hallway and parking lots. All nodes in the original bigraph, site 0 (from F_1 , a placeholder for the

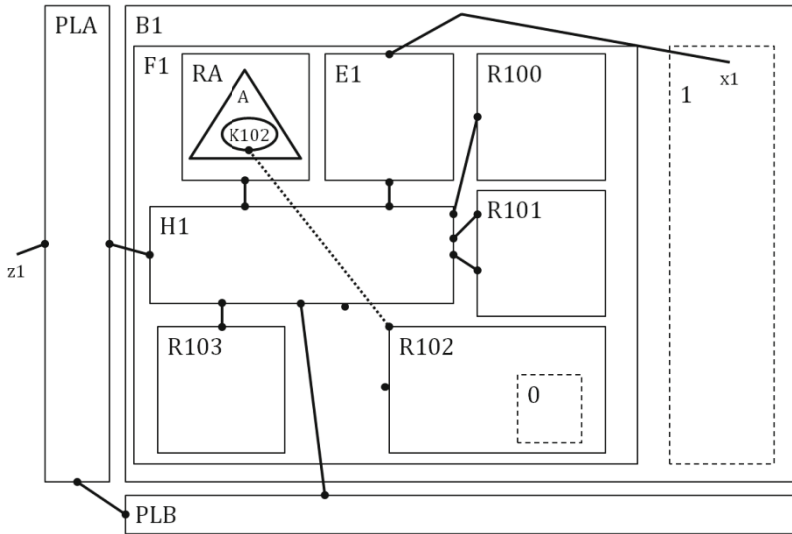


Fig. 10. Outdoor-Indoor bigraph $F1 \circ H: \langle \{0,1\}, \{x1\} \rangle \rightarrow \langle \{\}, \{z1\} \rangle$

contents of R102), and site 1 (from H, the placeholder for the 2nd floor) are left unchanged, as are open links $z1$ (an outer name indicating access to the outdoor road network) and $x1$ (an inner name indicating access via the elevator to the 2nd floor).

Resolving Inner Names and Sites

Suppose that the 2nd floor plan looks much like the 1st, except that there are no external exits and there is a bathroom in the upper left corner. Because we need to combine this bigraph with the rest of the building bigraph, we place the nodes inside a root 1 and include an open link (here an outer name) $x2$ from elevator E2 (Fig. 11).

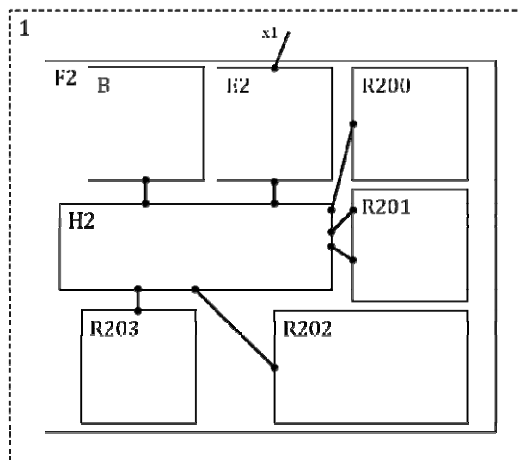


Fig. 11. 2nd floor bigraph $F2: \langle \{\}, \{\} \rangle \rightarrow \langle \{1\}, \{x1\} \rangle$

Composing the outdoor-indoor bigraph and F2 joins the building bigraph site 1 with F2’s root 1, and inner and outer names $x1$ (Fig. 12).

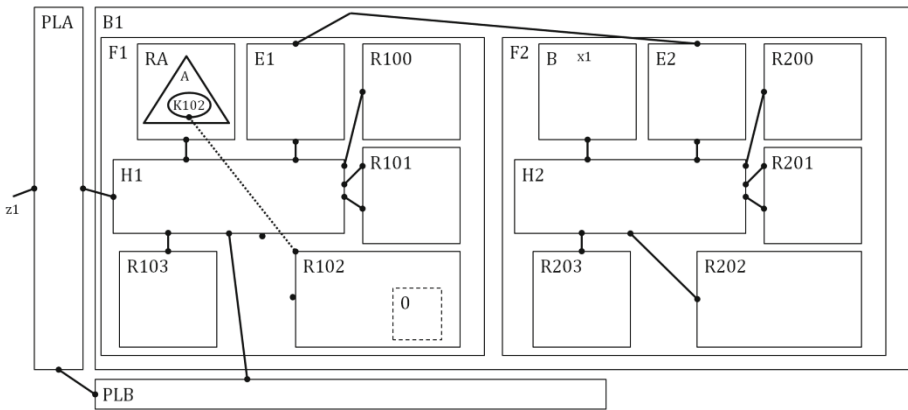


Fig. 12. Multistory indoor-outdoor bigraph $F2 \circ F1 \circ H: \langle \{0\}, \{ \} \rangle \rightarrow \langle \{ \}, \{z1\} \rangle$

Inner name $x1$ from the indoor-outdoor bigraph is joined with outer name $x1$ from F2 to form a new closed edge (access link) between elevator nodes, and site 1 in the indoor-outdoor bigraph has joined with root 1 in F2. Note that the nodes and closed links are left unchanged after bigraph composition.

5 Representing Change in Bigraphs

Bigraphs can be modified by the application of reaction rules, which specify legal changes to linking and place relations. A reaction rule consists of a pair of bigraph parts consisting of a *redex* (pattern to be changed) and *reactum* (resulting pattern). Most domains modeled with bigraphs require one or more reaction rules that support changing basic locality or linking relations. For indoor bigraphs agent actions such as going *into* or *out of* or *locking/unlocking* a place are modeled with rules. The choice of appropriate reaction rules that change a single placing or link relation based on spatial *image schemas* was explored in earlier work [7]. Here we provide a representative sample of reaction rules (associated with the CONTAINER and LINK schema respectively) for modifying indoor bigraphs in response to agent actions.

INTO Rule

An agent with a key may move into any place accessible from her current location, whether or not the rooms have additional content (Fig. 13). Open links on the nodes indicate that other links are permissible parts of the pattern to be matched. This rule is self-inverse, and no link relations are changed by applying it.

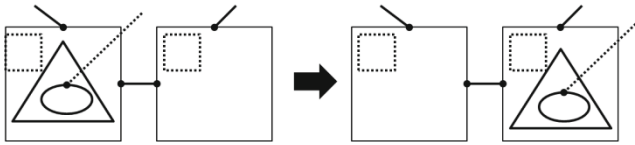


Fig. 13. INTO Rule: INTO(a,p) is the action of $a \in \text{Agents}$ moving into $p \in \text{Places}$

LINK (UNLOCK) Rule

Because we have modeled the relationship between a key and the room it unlocks as a link we require an access LINK rule (corresponding to an UNLOCK action) which creates an access link between places after a door is unlocked (Fig. 14). An inverse UNLINK (LOCK) rule is also usually required to make a place inaccessible [7]. No place relations are changed when applying this rule.

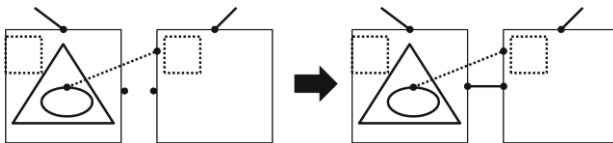


Fig. 14. LINK Rule: ALINK(p1,p2) is the action of $p1$ linking to $p2$ where $p1,p2 \in \text{Places}$

There are many rule variants. For example, with more specific agent and place types the INTO rule for stairways can be restricted to only allow agents that can use stairs to enter stairways. Therefore, a well-typed agent that can't use stairs wouldn't be sent into stairways because that action couldn't be modeled in the formal system.

5.1 Indoor Navigation

Combining bigraphs with a set of reaction rules yields a *bigraphical reactive system* (BRS), in which indoor navigation can be modeled by modifying bigraphs (temporal snapshots of indoor environments) by the sequential application of appropriate rules. For example, suppose the agent with a key in the reception area from Fig. 7 wishes to access locked room 102. We do not need information about the 2nd floor or the outdoors to model these actions. The first step is applying the INTO rule, resulting in a new bigraph showing the agent has moved from RA into the hallway (Fig. 15).

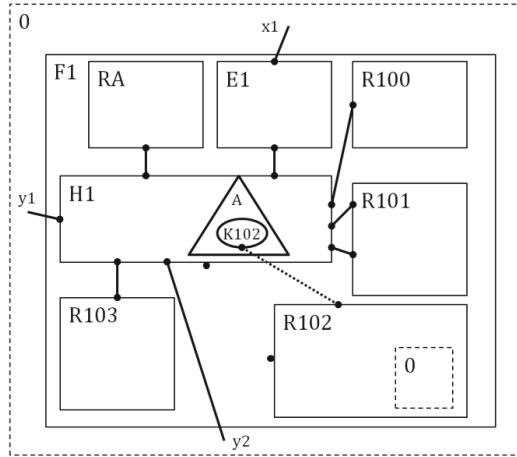


Fig. 15. F1 after applying the INTO rule

Next, use the LINK (UNLOCK) rule to change the link relation between the hallway and RM102 by connecting the open ports to make the room accessible (Fig. 16):

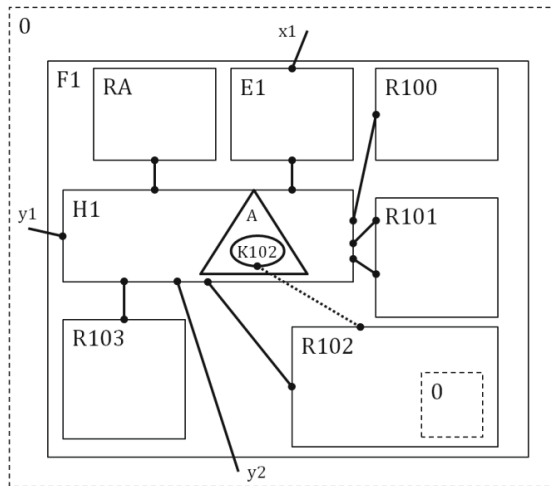


Fig. 16. Indoor scene after applying UNLOCK rule

Finally, use the INTO rule again to model the agent entering RM102 (Fig. 17).

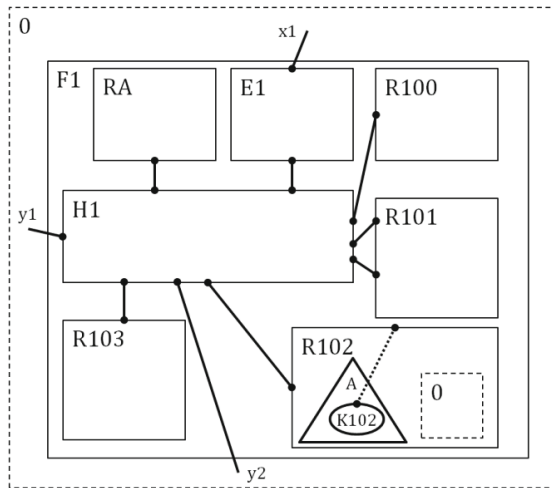


Fig. 17. Agent reaches her destination

By defining key indoor events (corresponding to atomic agent actions) and modeling them as reaction rules we can construct sequences that model dynamic agent behaviors to modify indoor environments. Given an agent's starting situation and a particular navigation task we can determine potential sequences of events that would lead to satisfying her goal (e.g., how can she reach the nearest bathroom?). This can be done even when there is incomplete information about the environment, such as not knowing where the building exits lead when the tasks involve actions on just one floor.

6 Conclusion and Future Work

This paper describes a qualitative framework for formally representing and reasoning about indoor environments to support indoor navigation tasks. Our primary goal was to demonstrate that indoor bigraphical models are appropriate formalization and visualization tools for indoor environments that support reasoning about the effects of indoor events (precipitated by agent actions) on key indoor environmental elements.

Indoor bigraphs provide formal algebraic specifications of indoor environments that independently represent agent and place locality (e.g., building hierarchies) and connectivity (e.g., path based navigation graphs). Our examples illustrate that indoor bigraphs can be constructed from building floor plans with some additional scene information (e.g., agent and mobile object locations). Further, we demonstrated that scenes with partial information (e.g., incomplete building plans) could be modeled in a way that supported adding additional context, including outdoor contexts, suggesting a new approach to integrating outdoor and indoor navigation systems.

In related work the authors have developed constructions that provide explicit bigraph types for representing complex two-dimensional spatial configurations [11],

and defined preliminary ontologies for indoor and hybrid outdoor-indoor spaces for built environments based on a typology of the space and the entities it contains [12].

This paper extends the authors' previous work on using image schemas to model spatial relations and agent actions in bigraphs [7]. To improve our representation of agent behaviors in indoor environments we also plan to incorporate *affordances* into the framework. Affordances describe objectively measurable actions an agent can take in an environment given their current capabilities [13]. Reaction rules in the original ambient calculus were established based on context-dependant abilities of certain ambients (processes) to perform actions [9]. Similarly, modeling affordances in indoor bigraphs should help to refine reaction rules and improve reasoning procedures in goal directed navigation task planning.

Future work will include the integration into the framework of an indoor *event calculus* [14] to provide a logic-based formalism for representing the *effects* of indoor events on indoor relationships. Currently, the effects of agent actions on indoor relationships are modeled with reaction rules (e.g., when an agent enters a room it has the effect of changing their location). By defining appropriate *effect axioms* in the calculus corresponding to the rules, we will be able to automatically generate time indexed *narratives* about indoor navigation tasks as sequences of events and their consequences in indoor environments. This will support forward reasoning (e.g., what sequence of actions can an agent take to reach a particular place from their current location?) and backwards (explanatory) reasoning (e.g., given that an agent is in a particular place now, how could they have gotten there from some previously known location?). Automated reasoning about indoor navigation using qualitative formal models has the potential to improve many kinds of spatial information systems such as providing decision support for visitors to large building complexes or analytic support for security personnel using alert systems to reconstruct possible security breaches involving unauthorized access to restricted areas or materials.

References

1. Howell, I., Batcheler, B.: Building Information Modeling Two Years Later – Huge Potential, Some Success and Several Limitations. *The Laiserin Letter* (2005)
2. Franz, G., Mallot, H., Wiener, J.: Graph-based Models of Space in Architecture and Cognitive Science - a Comparative Analysis. In: Leong, Y.-T., Lasker, G.E. (eds.) *Proceedings of the 17th International Conference on Systems Research, Informatics and Cybernetics*, pp. 30–38 (2005)
3. Lee, J., Kwan, M.P.: A Combinatorial Data Model for Representing Topological Relationships between 3-D Geographic Entities. *International Journal of Geographical Information Sciences* 19(10), 1039–1056 (2005)
4. Stoffel, E.-P., Lorenz, B., Ohlbach, H.J.: Towards a Semantic Spatial Model for Pedestrian Indoor Navigation. In: Hainaut, J.-L., Rundensteiner, E.A., Kirchberg, M., Bertolotto, M., Brochhausen, M., Chen, Y.-P.P., Cherfi, S.S.-S., Doerr, M., Han, H., Hartmann, S., Parsons, J., Poels, G., Rolland, C., Trujillo, J., Yu, E., Zimányi, E. (eds.) *ER Workshops 2007. LNCS*, vol. 4802, pp. 328–337. Springer, Heidelberg (2007)

5. Richter, K.-F., Winter, S., Rüetschi, U.-J.: Constructing Hierarchical Representations of Indoor Spaces. In: IEEE International Conference on Mobile Data Management, pp. 686–691. IEEE Computer Society, Los Alamitos (2009)
6. Milner, R.: The Space and Motion of Communicating Agents. Cambridge University Press (2009)
7. Walton, L., Worboys, M.: An Algebraic Approach to Image Schemas for Geographic Space. In: Hornsby, K.S., Claramunt, C., Denis, M., Ligozat, G. (eds.) COSIT 2009. LNCS, vol. 5756, pp. 357–370. Springer, Heidelberg (2009)
8. Milner, R.: Communicating and mobile systems: the π -calculus. Cambridge University Press, Cambridge (1999)
9. Cardelli, L., Gordon, A.: Mobile Ambients. Theoretical Computer Science, Special Issue on Coordination 240(1), 177–213 (2000)
10. Cardelli, L.: Abstractions for Mobile Computation. In: Vitek, J., Jensen, C.D. (eds.) Secure Internet Programming. LNCS, vol. 1603, pp. 51–94. Springer, Heidelberg (1999)
11. Worboys, M.: Using Bigraphs to model topological graphs embedded in orientable surfaces. Journal of Theoretical Computer Science (2010) (submitted)
12. Walton, L., Worboys, M.: Indoor Spatial Theory. Technical report presented at the ISA project meeting held at the 2010 International Workshop on Indoor Spatial Awareness, Taipei, Taiwan (2010)
13. Gibson, J.: The Theory of Affordances. In: Shaw, R., Bransford, J. (eds.) Perceiving, Acting, and Knowing (1977)
14. Kowalski, R.A., Sergot, M.J.: A Logic-Based Calculus of Events. New Generation Computing 4, 67–95 (1986)

Dynamic Refuse Collection Strategy Based on Adjacency Relationship between Euler Cycles

Toyohide Watanabe and Kosuke Yamamoto

Department of Systems and Social Informatics,
Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan
watanabe@is.nagoya-u.ac.jp

Abstract. Our objective is to reduce the risk of overwork in the refuse collection procedure while keeping efficient routes. On optimum routes in refuse collection, vehicles pass through each road segment only once. When we look upon our road network as a graph, the optimum route is Euler graph. Euler graph consists of several Euler cycles. When Euler cycles are exchanged in Euler graph, these cycles are yet Euler cycles if the exchanged cycles are adjacent. Our idea is to construct the cycle graph, which represents cycles as nodes and connective relationships between adjacent cycles as links, from Euler graph. It is guaranteed that the cycle based on links in the cycle graph does not generate the redundancy. In the computer simulation, we conclude that our method is effectively applicable to many kinds of road networks.

Keywords: Euler graph, cycle graph, refuse collection, combinational optimum problem.

1 Introduction

High cost of refuse collection work is regarded as a social problem because the volume and kinds of exhausted refuses increase day by day. It is very difficult to make up an effective collection plan to be applicable for several vehicles of refuse collection works under many complicated constraints such as the capacities of vehicles, the amount of refuses, the scales of daily events in roads, etc. As a result, the problem, that the amount of refuses oversupplies the limitation of predefined capacity, may occur as overwork. It is desirable to select the route which does not generate the overwork. However, it is very difficult to keep the effective routes so as to avoid overworking completely. In this paper, we address a flexible refuse collection method for attaining the effectiveness of refuse collection works and avoiding overworks.

The refuse collection problem is one of combinational optimization problems such as Capacitated Arc Routing Problem (CARP) [1]. CARP is NP-hard combinational optimization problem, and the strict solution is applicable to only an instance of strongly constrained problem. After Golden et al. formulated CARP in 1981, various types of solution methods such as path-scanning [2], Ulusoy's heuristics [3], etc. have been proposed, and also the researches which investigate meta-heuristics-like

methods based on genetic algorithm [4], tabu-search [5] and so on have been reported. In addition, Logo et al. proposed the means which transform CARP into Capacitated Vehicle Routing Problem (CVRP) [6] and compute the lower bounding value [7].

The uncertainty in estimating the amount of refuses is an important factor in our refuse collection problem. It is impossible to exactly predict the volume of refuses put in each road segment because the amount of exhausted refuses is variant every day. Since the collection routes are planned based on the predicted volume of already-exhausted refuses, the overworks may often occur in case that the predicted volume was different from the practically collected volume. Thus, the actively applied strategy is planned with surplus capacities in comparison with the maximum exhausted refuse volume. The ordinary method is designed so as to construct the collection route by vehicles with capacities which can carry out a large amount of refuses even if the volume were too much, but the estimation is redundant and is not effective. Fleury et al. defined stochastic CARP (SCARP) as a special case of CARP in which the collections of demands are changeable on every trial, and proposed SMA (Stochastic MA) [8] as Memetic algorithm [9] based on the objective function which depends on the change of demands. They computed the collection volume in SCARP less than that in the practical case, then applied the robust routing to the collection work even if the capacities were changed, and analyzed in detail for efficient routing on the basis of these processes [10]. Their method may be similar to the procedure used practically in our Nagoya city, whose policy keeps the accumulated capacity for the maximum predictable situation, but is not always satisfied with every occurrence. The redundancy for establishing various types of possible cases is too much loss.

2 Dynamic Refuse Collection Problem

The static refuse collection procedure is not sufficient to make cost-effective plan without overworks in the environment where the volume of locally exhausted refuses is changeable day by day. It is necessary to establish the framework for formulating flexible routing problem on the basis of the practically accumulated refuse volume with a view to collecting refuses without overworks. We discuss a dynamic refuse collection problem, using the idea that the routes based on Euler graph can be adaptively exchanged even on the half way of preset route.

2.1 Routing and Euler Graph

We regard that CARP is one of special allocation problems which do not only compute optimal routes but also look upon edges with the corresponding demands as tasks. A set of trips, which is a solution in CARP, is computed as a collection of cycles which connect among tasks. In this viewpoint, we can regard that the shorter the redundantly selected routes in each trip is, the more effective the route is. Euler cycle is defined as a closed path whose all edges are not redundantly constructed. The graph G is called as Euler graph when there are Euler cycles for all edges in G . We can look upon edges, which redundantly connect in order to pass through all edges even if they were not Euler graphs, as virtually constructed edges. In Figure 1, though

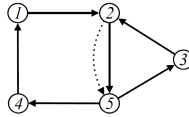


Fig. 1. Virtual Euler graph

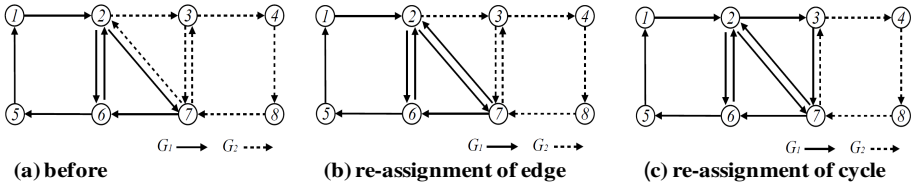


Fig. 2. Example of graph update

the real line segments do not compose an Euler graph, an Euler cycle is virtually composed with the broken line segment: the path “1→2→5→3→2→5→4→1” is an Euler graph.

We define the minimum virtual Euler graph whose added edges all have totally the minimum cost. In this case, the trip update procedure keeps the minimum virtual Euler graph so as to reconstruct Euler graphs. In an Euler graph, the following theorem is definitely established:

【Theorem 1】

If and only if the graph G is an Euler graph, a set of edges in G is dividable into one or more cycles. ■

This theorem insists that a graph is composed of only one cycle without duplicated paths and the cycle is a minimum Euler graph. When an Euler graph has two or more cycles, these cycles are always adjacent to one of other cycles mutually. When a trip must be updated, tasks (i.e., edges attached with demands) contained in each trip are necessarily exchanged. Generally the permutation of single edges between Euler graphs transforms Euler graph to un-Euler graph. Theorem 1 makes it clear that Euler graph G keeps its own properties as long as the connectivity is assured even if a new cycle was combined to G or the existing cycle was removed from G .

Figure 2(a) shows two adjacent directed graphs G_1 and G_2 : the edges in G_1 are indicated by real arcs, and the edges in G_2 are represented by broken arcs. Here, G_1 and G_2 are both Euler graphs because in G_1 the path “1→2→6→2→7→6→5→1” is observed; and in G_2 the path “2→3→7→3→4→8→7→2” is so. Consider a graph update procedure by reassigning edges in G_2 into those in G_1 . In reassigning the edge (7,2) the updated graph is illustrated in Fig.2(b). In Fig.2(b), G_1 and G_2 are not Euler graphs; while, the graph is shown in Fig.2(c) when the edge (2,3,7) is reassigned. In Fig.2(c), G_1 and G_2 are Euler graphs because the path “1→2→7→6→2→3→7→2→6→5→1” in G_1 and the path “3→4→8→7→3” in G_2 are Euler cycles.

2.2 Reassignment of Cycles

We propose a dynamic trip update algorithm by the cycle reassignment, based on the feature of Euler graph. First, we generate virtually Euler graph from the road network, and divide it into several cycles. Mourao et al. proposed a heuristic method that extracts cycles from a graph as a solution of static CARP, and then generates several Euler graphs by combining them [11]. Basically, we extract the cycles by means of Mourao’s method and generate an initial plan by assigning them to each trip. In our refuse collection process, we suppress the overwork by exchanging neighboring cycles: the cycle in overworked trip is exchanged by the un-full cycle.

Additionally, we take care of redundancy in a refuse collection plan so as to keep time-effectiveness: (1) to go back the route in order to pass through newly assigned cycles; and (2) to reassign the cycle, which contains in already processed task, to another trip. It is necessary to concentrate on the framework for being responsible to the time-variant situation and keeping route-effectiveness under the reassigned path pattern. To attain this viewpoint, we introduce a “cycle graph”, which represents the adjacent relationship between cycles as a link. The link in the cycle graph is erased under a predefined condition along vehicle movement. This mechanism is sure that the plan does not have redundancy when the cycle was reassigned with respect to the neighboring relationships denoted by links.

3 Mixed SCARP

We define MSCARP (Mixed SCARP) on the basis of SCARP. In MSCARP, two different demands such as real demand and predicted demand are defined in each edge. The real demand is the corresponding demand in CARP: the demand in CARP has a static value, and the real demand in MSCARP holds a probable value. We cannot know the real demand of edge which each vehicle does not yet visit. On the other hand, the predicted demand is a static and predictable value, derived from the statistical distribution of real demands, and is known in advance. In our refuse collection problem, the real demand is the amount of practically exhausted refuses, and the predicted demand represents the predictive amount of refuses, computed by using statistical information gathered during a constant period.

Now, we define our MSCARP formally. When the mixed graph G is given, the formula is:

$$\begin{aligned}
 G &= (N, A \cup E) && (1) \\
 N &= \{ n_o, n_1, \dots, n_M \} \\
 A &= \{ a_{ij} \} && (n_i \in N, n_j \in N) \\
 E &= \{ e_{ij} \} && (n_i \in N, n_j \in N) \\
 G^R &= \{ N, A^R \subseteq A \cup E^R \subseteq E \} \\
 a_{ij} \in A^R &= (c_{ij}^d, c_{ij}^s, q_{ij}^p, q_{ij}^r) \\
 a_{ij} \in A/A^R &= (c_{ij}^d) \\
 e_{ij} \in E^R &= (c_{ij}^d, c_{ij}^s, q_{ij}^p, q_{ij}^r) \\
 e_{ij} \in E/E^R &= (c_{ij}^d)
 \end{aligned}$$

Here, N is a set of M nodes, $a_{ij} \in A$ is a directed edge (or arc) from n_i to n_j , $e_{ij} \in E$ is an un-directed edge (or edge) between n_i and n_j . G is composed abstractly from the practical road network, N is a set of intersections, and $A \cup E$ is a set of road segments between intersections. In bi-directional road segments, the road segment, in which the refuse collection procedure can complete in one-way passing, is represented by un-directed edges; and one in which the collection procedure must complete in 2-ways passing is represented by two arcs. In case that two different nodes are combined by an arc, the edge means one-way road segment. Here, consider that G^R is a partial graph of G , and that A^R and E^R are, respectively, sets of arcs and edges with their own demands. $a_{ij} \in A^R$ and $e_{ij} \in E^R$ have individually pass cost c_{ij}^d , service cost c_{ij}^s , predicted demand q_{ij}^p and real demand q_{ij}^r as their own attributes. Only pass cost c_{ij}^d is assigned as their attributes to $a_{ij} \in A/A^R$ and $e_{ij} \in E/E^R$. The pass cost c_{ij}^d is a cost to be exhausted when the vehicle passed through the corresponding link and the service cost c_{ij}^s is a cost when the vehicle processes the demand preset on the road segments. The predicated demand q_{ij}^p is computed probably on the basis of normal distribution $N(q_{ij}^p, \alpha^2, (q_{ij}^p)^2)$. α is a positive constant and the standard distribution is $\alpha * q_{ij}^p$.

We represent the vehicle $v_k \in V$ and the trip $t_m \in T$. Also, the capacity of vehicle v_k is represented as cap_k . Here, the demand accumulated from each vehicle is represented as $load(v_k)$. The amount of currently accumulated refuses can be reset to 0 ($=load(v_k)$) with a constant sweep-out cost c^{demp} at the particular node “depot”. When such a vehicle ($v_k \in V$) starts the routing work, “ $load(v_k)=0$ ” as the initial value. Under such a circumstance, each vehicle v_k exhausts service cost c^s and increases $load(v_k)$ by real demand q^r whenever v_k processes the demands associated with the corresponding arcs/edges. The arc/edge, in which demands have been already gotten rid of, will be deleted from $A^R \cup E^R$. If $load(v_k) \supset cap_k$ in individual vehicles v_k 's, the collection process will be finished as the overwork happened.

4 Search Algorithm

In this section, we describe a dynamic routing algorithm for MSCARP. The generation of initial trip set is based on Mourao’s method [11], but we make the trip update operation flexible by keeping the adjacent relationship among cycles. Figure 3 is our processing flow.

4.1 Initial Routing

We transform the input graph G to a virtual Euler graph with a view to constructing Euler cycles from each trip. By Mourao’s method, a virtual directed Euler graph is generated after having transformed mixed graph into directed graph. Here, the condition that the directed graph G^D is equally an Euler graph means that the number of inputs is equal to the number of outputs in all nodes in G^D . The number of outputs is the number of arcs to exit from the corresponding nodes; and the number of inputs is the number of arcs to enter into the corresponding nodes.

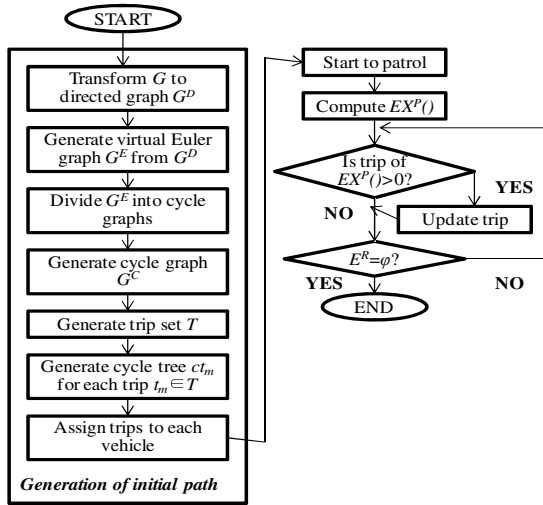


Fig. 3. Processing flow

Next, we generate the virtual Euler graph G^E from G^D . The important point is to construct virtual edges between nodes when the number of input nodes is different from the number of output nodes. Since these virtual arcs are redundant road segments, which enable vehicles to pass through the same road segments redundantly, it is desirable to minimize the total collection cost. This minimum problem of total collection cost is looked upon as Transportation Problem (TP) because the node whose number of input arrows is more than the number of output arrows is a producer, and the node whose number of output arrows is more than the number of input arrows is a consumer [12]. To solve TP is to minimize the total transportation cost when the transportation costs are given under individual terms: producers and consumers; the producer volume and consumer volume for some products; combination between producers and consumers; and so on. Figure 4 shows examples of virtual arcs. The number, attended to left-upper side of each node, is a computation value of “number of input arcs – number of output arcs” for the corresponding node. In Fig.4(a), the number of input arcs is by 1 more than the number of output arcs between nodes n_4 and n_5 ; and the number of output arcs is by 1 more than the number of input arcs between nodes n_2 and n_3 . Thus, the graph in Fig.4(a) is not an Euler graph. If virtual arcs are added between $a_{4,2}$ and $a_{5,3}$, the graph is shown in Fig.4(b). In this case, this graph is an Euler graph as “(number of input arcs – number of output arcs) = 0” in all nodes. Virtual arcs, denoted by broken line segments, are the minimum paths to connect between end sides of both nodes. In Fig.4(b), $a_{4,2}$ and $a_{5,3}$ construct paths “ $n_4 \rightarrow n_5 \rightarrow n_2$ ” and “ $n_5 \rightarrow n_2 \rightarrow n_3$ ”, when the pass costs of arcs, represented by the real line segments, are at all equal.

When the directed graph G^D is transformed into the Euler graph G^E , all arcs on a path constructed by virtual arcs must be copied. In this case, these copied arcs do not associate demands q^r and q^p , and service cost c^s . Using a set of copied arcs A^{TP} , G^E is expressed in Expression (2):

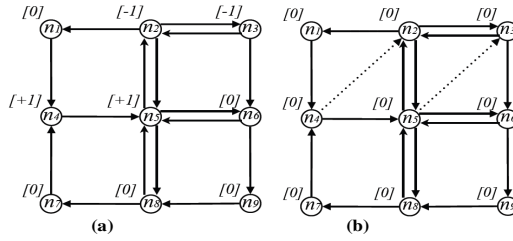


Fig. 4. Example of adding virtual arcs

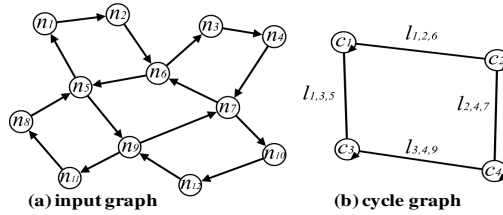


Fig. 5. Example of cycle graph

$$G^E = (N, A^E) \tag{2}$$

$$A^E = A^D \cup A^{TP}$$

We divide G^E into several cycles; each cycle c_m is looked upon as a partial graph of G^E ,

$$c_m = (N^C_m \subseteq N, A^C_m \subseteq A^E) \tag{3}$$

Moreover, we define the total sum of predicted demands over the cycle c_m as a predicted demand of c_m :

$$dem^p(c_m) = \sum_{aij \in A^C_m \cap A^R} q^p_{ij} \tag{4}$$

A set of cycles in G^E is partitioned as follows:

- (1) Initialize the set of cycles C by \varnothing ;
- (2) Select a cycle c_i ($dem^p(c_i) > 0$) whose predicted demand $dem^p(c_i)$ is minimum when there is the cycle, started from n_i in G^E for $\forall n_i \in N$;
- (3) Select maximum $dem^p(c_i)$ from cycle $\{c_i\}$, generated in (2);
- (4) Erase arcs in the corresponding cycles from A^E after having inserted cycles in (3) into C ;
- (5) Finish if $A^D \cap A^R = \varnothing$; Otherwise, goto (1).

These steps generate a set of cycles whose predicted demands are small in average. Thus, since the change of demand in each trip derived from the reassignment of one cycle is not many, more flexible routing alteration procedure becomes possible.

After the division of cycles in the Euler graph finished, a set of trips is generated by the combination of neighboring cycles. This combination is effective under the condition that the total sum of predicted demands in individual trips does not overwork the allowable demands of vehicle. We select the cycle to be combined on the basis of the resolution possibility with respect to Mourato's method. Our trip is constructed by adding the shortest round-trip path for depot into the graph, derived from the combination of cycles. Each vehicle selects a cycle at random from un-round trip, and repeats this process.

4.2 Cycle Graph

The cycle graph is a graph which represents the neighboring relationship between cycles as link and the cycle as node, and is defined as

$$\begin{aligned}
 G^C &= (C, L) \\
 C &= \{ c_0, c_1, \dots \} \\
 L &= \{ l_{ijk}, \dots \} \\
 l_{ijk} &= (c_i, c_j, n_k)
 \end{aligned}
 \tag{5}$$

Here, C is a set of cycles and L is a set of links. The link $l_{ijk} \in L$ uniquely exists for the combination of two cycles $c_i, c_j \in C$ with neighboring relationship, and node n_k , assigned to these cycles. Figure 5 is an example. The cycle graph is shown in Fig.5(b) when the directed Euler graph in Fig.5(a) was divided into cycles c_m ($m=1,2,3,4$) of four nodes $\{n_1, n_2, n_6, n_5\}, \{n_3, n_4, n_7, n_6\}, \{n_5, n_9, n_{11}, n_8\}$ and $\{n_7, n_{10}, n_{12}, n_9\}$. The link $l_{1,2,6}$, which connects to c_1 and c_2 in Fig.5(b), represents that the node n_6 is shared between c_1 and c_2 . Similarly, other links are so.

The trip is constructed by adding the shortest round-trip path for depot into the mutually adjacent cycles. In this case, the combination part of cycle is a sub-graph of cycle graph. Here, we define a partial graph $sg^C(t_m)$ of cycle graph corresponded to trip t_m as follows:

$$sg^C(t_m) = (C'_m \subseteq C, L'_m \subseteq L)
 \tag{6}$$

We call the link in cycle graph, which connects cycles in the same cycle, **inner link**, and the link in cycle graph, which connects cycles in the different cycles, **external link**. We express a set of inner links as L^{IN} , and a set of external links as L^{OUT} . Figure 6 is an example of partial graphs $sg^C(t_1), sg^C(t_2)$ for two trips t_1 and t_2 :

$$\begin{aligned}
 C'_1 &= \{ c_1, c_2, c_3, c_4, c_5 \} \\
 C'_2 &= \{ c_6, c_7, c_8, c_9 \}
 \end{aligned}
 \tag{7}$$

Since the connective partial graph in the cycle graph G^C represents an Euler graph, it is necessary to establish the connectivity of $sg^C(t_m)$ with a view to keeping the time-effective cost. As the cycles, which have been already started to collect refuses, cannot be erased from the already-assigned trip even if they had been allocated to other trips, the redundancy were generated undesirably, and the assigned trip and assigning trip contained the cycle mutually at once. Under the above consideration, we define the conditions for reassignment of cycles as follows:

[Definition: Condition for reassigning cycle]

- (1) external link is connected;
- (2) cut-off point is not set in cycle graph, corresponded to trip;
- (3) collection is not started yet.

Figure 7 is an example, derived from the road network shown in Fig.6: the cycle in trip t_1 is reassigned into trip t_2 , and these trips are updated. Though Fig.7(a) is an example which looks upon the cycle c_1 as reassignment target, this does not satisfy with condition 1. Like this example, the trip attended with reassignment target may become un-connective when the cycle which is not connected to external link was reassigned. On the other hand, the condition 2 is not satisfied though Fig.7(b) is an example in which the cycle c_4 was looked upon as reassignment target. The cut-off point is a node which makes the corresponding graph disjoint when it was gotten rid of. Fig.7(c) is an example in which the cycle c_2 is reassigned and satisfies conditions 1 and 2. However, if the collection operation for c_2 has already started this updated pattern is not allowed concerning to the condition 3.

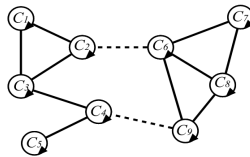


Fig. 6. Inner links and external links

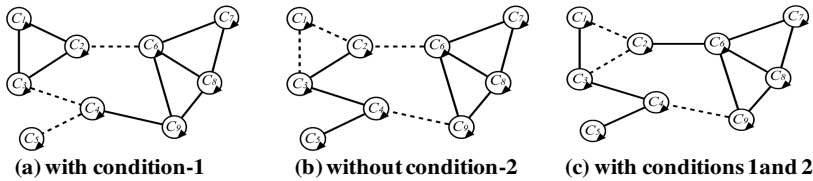


Fig. 7. Example of trip update

4.3 Cycle Tree

Trips are constructed as Euler graphs. It is necessary to trace all Euler cycles which cover through all arcs in the given Euler graph without any duplication so that all arcs in a trip can be assigned to each vehicle. Such an Euler cycle ec is generated from a partial graph $sg^C(t_m)$ of cycle graph, assigned to each trip, as follows:

- (1) Select start node n_0 and start cycle $c_0 \in C_m^t$. Here, n_0 is included in a set of nodes for c_0 ;
- (2) Initialize ec as a cycle, corresponded to c_0 after starting from n_0 ;

- (3) Select a cycle c_m which connects to ec and is not included in ec as for any node n_i in ec ;
- (4) Insert cycle, surrounded in c_m from n_i to ec ;
- (5) Finish if ec includes all cycles in $sg^C()$; Otherwise, goto (3).

A set of links, which represents neighboring relationship used to select the cycle in step 3, is a set of edges in global tree of cycle graph G^C . We call such a global tree the cycle tree, here.

Examples are shown in Figure 8, Figure 9, and Figure 10. Fig.8(a) shows a directed Euler graph G^E , and Fig.8(b) shows an example of cycle graph G^C derived from G^E . Fig.9 illustrates the transition sequence of Euler cycle ec when the above operations have been repeated sequentially in G^E . The grey symbols on arcs indicate the order of ec . Fig.10 is the result when we applied these operations to the cycle graph G^C : grey nodes are basically contained in ec and real line segments represent the neighboring relationship used in step 3. In Fig.9 and Fig.10, the figures in (a)-(e) have their mutual correspondences. ec becomes Fig.9(a) when c_1 is selected as the start cycle c_0 and n_1 is chosen as the start node n_0 . Next, ec becomes Fig.9(b) when the cycle c_2 is inserted by regarding n_2 as the start node. In this case, the neighboring relationship between cycle c_2 and partial graph, which was passed through ec , is indicated by the link $l_{1,2,2}$ on G^E . $l_{1,2,2}$ is depicted by real line segment in Fig.10(b). When we insert into ec by the order “ c_5 as the start node n_7 ; c_4 as the start node n_5 ; c_3 as the start node n_9 ” ec is updated in the order such as Fig.9(c), Fig.9(d), and Fig.9(e). They are represented in G^C by the order of Fig.10(c), Fig.10(d), and Fig.10(e). In Fig.9(e), the procedure is finished because ec has all cycles in G^C . In Fig.10(e) partial graphs generated by the links with real line segments are a global tree in G^C .

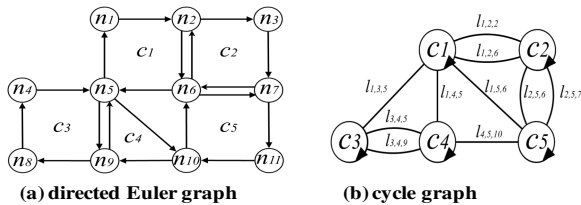


Fig. 8. Example of directed Euler graph and cycle graph

Our method generates the cycle tree on partial graph $sg^C_m(t_m)$ of cycle graph for each trip t_m after a set T of trips has been generated. Expression (8) shows a cycle tree ct_m for trip t_m :

$$ct_m = (C_m^t, L_m^b \subseteq L_m^t, c_m^{root}) \tag{8}$$

Here, $L_m^b \subseteq L_m^t$ is a set of links which were selected as edges of cycle tree ct_m from the inner link in the trip t_m . Also, c_m^{root} is the root cycle in a cycle tree and corresponds to the start cycle c_0 in the generation of Euler cycle. The collection of cycle tree is started

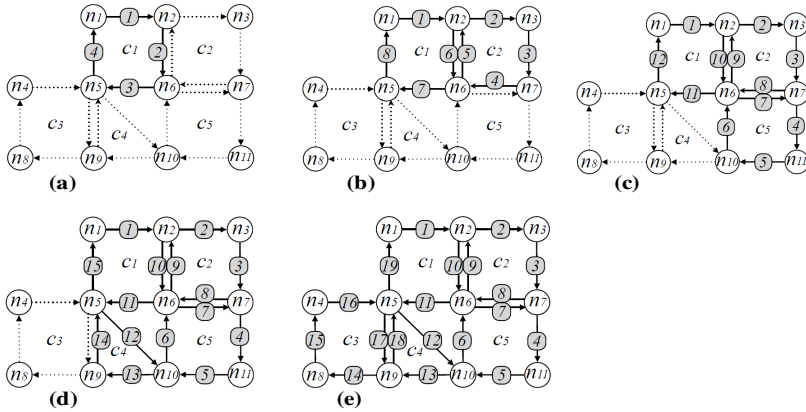


Fig. 9. Example of constructing Euler cycle

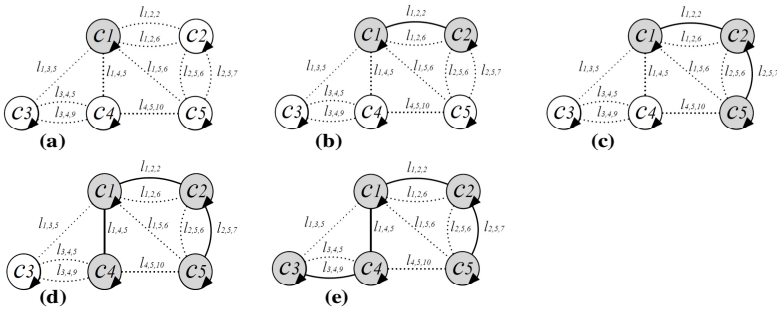


Fig. 10. Example of constructing cycle tree

from the root cycle c^{root}_m and the cycle, which contains the nearest node from depot, is desirably selected as c^{root}_m . Two features are observed by planning the collection order of vehicles in the cycle tree: it is easy to update the trip which satisfies constraints with time-effective management; and it is possible to optimize the collection order even by changing tree structure.

Euler path from the current position of vehicle to the depot must be constructed in the trip update operation in order to avoid the redundancy of route. Here, Euler path is defined as the path whose edges are only once passed through. Since the cycle tree ct_m corresponds to the Euler cycle of partial graph in G^E , indicated by $sg^C(t_m)$, it is sure that if the vehicle collects according to ct_m Euler path is always passed. Thus, if the cycle tree was constructed in advance, it is not necessary to construct Euler path initially as only the simple update of cycle tree is allowable even in the trip update. When the trip t_m was updated, the update procedure of ct_m is:

Step 1: insert into ct_m : the cycle tree c_k as leaf and the external link l_{ijk} as edge, in case that c_k was added by l_{ijk} ;

Step 2: delete c_k and edges, connected to c_k , from ct_m , in case that the allocation of cycle $c_k \in C_m$ to other trips is executed again;

Step 3: add $l \in L_m^l/L_m^b$, which connects between disjointed partial trees, into ct_m , in case that the cycles deleted in Step 2 are not leaves;

4.4 Trip Update

During the refuse collection, the overworks in individual trips are monitored. The overwork means that the amount of currently collected refuses is more than the finally predicted volume. The overwork in the trip t_m is $EX^p(t_m, v_k)$ if the vehicle v_k is in the collection phase; Otherwise, $EX^p(t_m)$ is for t_m :

$$EX^p(t_m, v_k) = load(v_k) + dem^p(t_m) - cap_k \tag{9}$$

$$EX^p(t_m) = dem^p(t_m) - cap_k \tag{10}$$

Here, $dem^p(t_m)$ is the sum of predicted demands in t_m , and is expressed as follows:

$$dem^p(t_m) = \sum_{a_{ij} \in A(t_m) \cap A^R} q^p_{ij} \tag{11}$$

The arc $a_{ij} \in A^R$, whose demand process has already finished, is deleted from A^R , and $load(v_k)$ increases by the real demand q^r_{ij} of a_{ij} . Thus, $dem^p(t_m)$ means the volume of predicted demands remained in t_m . Also, $EX^p(\)$ becomes the difference between the volume and a sum of real demands since $load(v_k)$ increases and $dem^p(t_m)$ decreases. When the value of overwork is positive in the trip, we must check the situation and re-plan the trip.

The processing flow in the trip update is shown in Figure 11. First, the counter *count* for reassignment number is set to 0 and the update target trip list T^{cand} is initialized by all trips T . The trip t_m is selected so as to maximize the overwork $EX^p(\)$ in T^{cand} . The cycle which satisfies reassignment condition is regarded as C^{sat}_m from a set of cycles C^c_m , contained in the partial graph $sg^C(t_m)$ of the cycle graph in t_m . It is not sure that there are constantly cycles which satisfy the conditions: if $C^{sat}_m = \varnothing$, t_m is deleted from T^{cand} , and other candidate trips are selected. If candidate trips cannot be found out, the operation finishes as failure of reassignment. Of course, if C^{sat}_m satisfies the condition of reassignment, the reassignment target cycle c^{ra} is always selected from the cycles. In this case, the cycle whose predicted demand $dem^p(c)$ is nearest to the overwork $EX^p(\)$ is selected. This is because a means makes the operation finish as soon as possible. Next, the reassignment trip $t^{ra} \in T/t_m$ of c^{ra} is selected at random from the cycles which are adjacent to c^{ra} . c^{ra} is assigned to t^{ra} again. In this case, the cycle tree is updated at once. If all trips of $EX^p(\) > 0$ were exhausted by the reassignment procedure, the operation is finished. Otherwise, *count* is incremented by 1; T^{cand} is again initialized by all trips. The operation is continued. However, when all trips did not satisfy $EX^p(\) \leq 0$ and *count* reached to the constant σ , the reassignment is regarded as failure and the procedure finishes.

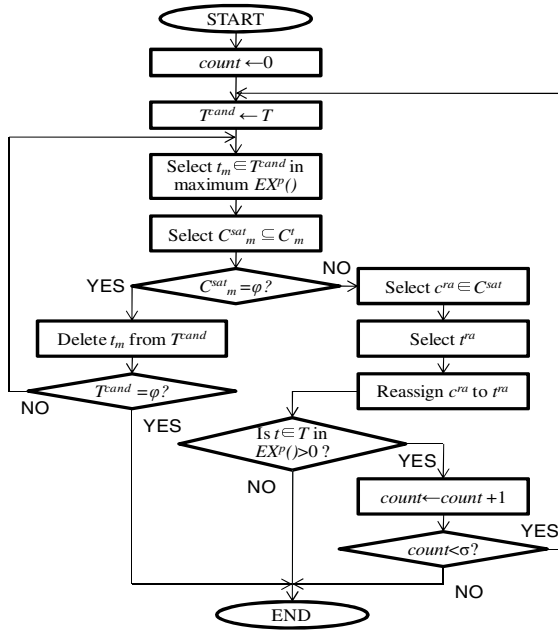


Fig. 11. Trip update

4.5 Optimization of Collection Sequence

In order to avoid the overwork in dynamic refuse collection, the collection plan, which makes it possible to update trips every time, is desirably composed. $EX^p()$ can be accordingly evaluated as reliable information with the difference between the capacity and practical demand in the latter phase of collection operation. Since the trip is updated on the basis of $EX^p()$, we can regard as the flexible collection plan when the possibility of trip update is kept high until the final point. The possibility in our trip update method is regarded as the number of cycles. Thus, in order to keep the possibility of trip update continuously until the final stage, it is better to decrease smoothly the cycles which satisfy reassignment conditions.

To compute the path in which the reassignment target cycles decrease smoothly, we define the evaluation value $EF(ct_m)$ of cycle tree ct_m as follows:

$$\begin{aligned}
 EF(ct_m) = & W^{OL} \sum_{l \in LOUT} (ltime^D_m(l)) + W^L \sum_{l \in LIN} (ltime^D_m(l)) \\
 & + W^C \sum_{c \in C} (ltime^V_m(c))
 \end{aligned} \tag{12}$$

W^{OL} , W^L , and W^C are weight values of number of external links, number of inner links and number of un-visited cycles. Also, $time^D_m(l)$ represents the time-cost in which link l in the cycle tree is deleted from the start of collection work when the vehicle works independently according to ct_m . $time^V_m(c)$ indicates the time period that the vehicle at first reaches at the cycle c from the start of work.

We optimize the cycle tree ct_m in order to compute the optimal collection order with respect to the evaluation function $EF(ct_m)$. Namely, we solve a sub-problem which computes ct_m to minimize $EF(ct_m)$ concerning to a trip t_m and then optimizes the collection plan based on the collection order. Our cycle tree is a global tree, which is derived from the partial graph of cycle graph included in the trip. When the number of nodes in the graph is lv , the number of edges included in the global tree is $lv-1$. In this case, the number of combinations with all edges is ${}_a\mathbf{C}_b$ (e.g. $a=|L^I m|$, $b=|C^I m|-1$) with respect to the partial graph $sg^C(t_m) = (C^I_m, L^I_m)$ corresponding to t_m in cycle graph. However, all search means of this combination are not realistic when the size is too larger. Concerning to the global tree, the minimum global problem [13] has been investigated, and Classical method, Prim method, etc. are well known as effective solution means.

5 Computer Simulation and Evaluation

We experimented by computer simulation and evaluated the experimental results in order to attain the subject “effective routing and overwork avoidance”. Our method assumes that the effective cost is not invariant even after the update because the trip update process is constrained by the cycle graph. Using the same initial route, we evaluated how many risks of overworks can be avoided/decreased by comparing the occurrence probability of overworks in the updated trip with that in the un-changed trips.

5.1 Experimental Setting

In our experiment, we used **lpr**, which Belenguer et al. have published as mixed-CARP standard data set [14]. **lpr** is a set of graphs, which organized virtually road networks for collecting refuses in local governments, and consists of three groups of datasets: **lpr-a**, **lpr-b** and **lpr-c**. **lpr-a** is a model for modern-type of city where each road is comparatively wide in bi-directional ways. Namely, this group has many pairs of arcs in bi-directional connection between two neighboring nodes. **lpr-b** is a model for old-type of city in which graphs in this group are in one-way: all arcs between two nodes are in a uni-directional connection. **lpr-c** is a model for urban-type of village without heavy traffics and consists of narrow-type in bi-directional paths. Each group contains five kinds of instances which are distinguished by the numbers 1-5. The more the number of instances is, the larger the size of graph is. The cycle tree is generated by ILS, using an initial solution computed by the breath-first search for the cycles being nearest to depot, and the weight in the evaluation function $EF(ct_m)$ is set to $1/3$.

5.2 Experimental Result

Figure 12, Figure 13 and Figure 14 show experimental results. Vertical axis indicates the occurrence ratio of overworks, and the horizontal axis is the distribution ratio α (=

0.0 -- 1.40) of real demands. Each diagram indicates the ratio that the collection work was finished by overworks in 200 trials: the real line segments without plotting points represent a set of static trips without update operation, and the line segments with plotting points represent our result.

In the static trip, we can observe that the more the sizes of instances in all groups **lpr-a**, **lpr-b**

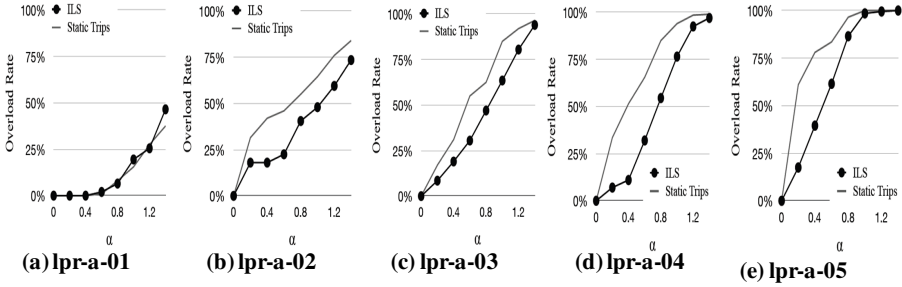


Fig. 12. Experimental result: lpr-a

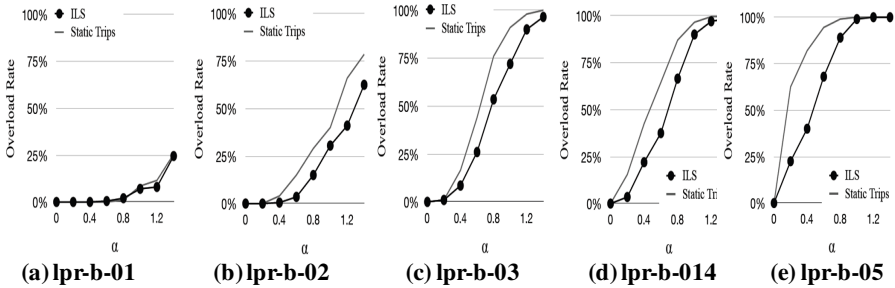


Fig. 13. Experimental result: lpr-b

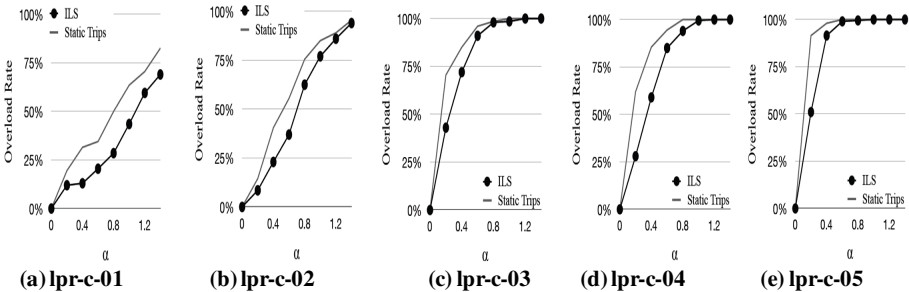


Fig. 14. Experimental result: lpr-c

and **lpr-c** are, the larger the slopes of individual graphs are. Namely, the occurrence ratio of overworks, attended with α , becomes large owing to the size of problem. Of

course, in our method when α is small, this increasing trend is reduced. Many results in our method are improved in comparison with those in the static trips. Also, the large difference between our method and the static trip method is observed in the scope that α is small in the large size of instances. In this observation **lpr-a-01** and **lpr-b-01** are not almost improved. Our method could not apply well, we think, because these are instances in the small size and the means for trip update are very limited at the start of collection work. Moreover, when we focus on the large size of instances, we can acquire good results for groups **lpr-a** and **lpr-b**, in particular.

6 Conclusion

In this paper, we proposed the dynamic refuse collection algorithm with cycle reassignment in order to select effective collection routes and avoid overwork from a viewpoint of the uncertainty of exhausted volume in the refuse collection. From our evaluation and experiment, we made it clear that the risk of overworks can be well controlled in accordance with the feature of road network and the distribution of exhausted refuses. In our experimental results, it is successful to reduce the risk of overworks in case of bi-directional way, wide road, and many one-way roads. Of course, it is hopeful to improve the performance by transforming edges into arcs even if the graph is composed of undirected edges in the initial routing step. In this experiment, we evaluated the risk of overworks, but it is necessary for us to investigate the framework under the constraints, which the overwork does not occur in order to apply our method to practical cases. As our future work, it is important and necessary to deal with such a problem.

References

- [1] Golden, B.L., Wong, R.T.: Capacitated Arc Routing Problems. *Networks* 11, 305–315 (1981)
- [2] Golden, B.L., Dearmon, J.S., Baker, E.K.: Computational Experiments with Algorithms for a Class of Routing Problems. *Computers and Operations Research* 10(1), 47–59 (1983)
- [3] Ulusoy, G.: The Fleet Size and Mix Problem for Capacitated Arc Routing. *European Journal of Operational Research* 22(3), 329–337 (1985)
- [4] Lacomme, P., Prins, C., Ramdane-Chérif, W.: A Genetic Algorithm for the Capacitated Arc Routing Problem and Its Extensions. In: Boers, E.J.W., Gottlieb, J., Lanzi, P.L., Smith, R.E., Cagnoni, S., Hart, E., Raidl, G.R., Tijink, H. (eds.) *EvoIASP 2001, EvoWorkshops 2001, EvoFlight 2001, EvoSTIM 2001, EvoCOP 2001, and EvoLearn 2001*. LNCS, vol. 2037, pp. 473–478. Springer, Heidelberg (2001)
- [5] Hertz, A., Laporte, G., Mittaz, M.: A Tabu Search Heuristic for the Capacitated Arc Routing Problem. *Operations Research* 48(1), 129–135 (2000)
- [6] Ralphs, T., Kopman, L., Pulleyblank, W., Trotter, L.: On the Capacitated Vehicle Routing Problem. *Mathematical Programming* 94(2/3), 343–359 (2003)
- [7] Longo, H., de Aragao, M.P., Uchoac, E.: Solving Capacitated Arc Routing Problems Using a Transformation to the CVRP. *Computers and Operations Research* 33, 1823–1837 (2006)

- [8] Fleury, G., Lacomme, P., Prins, C.: Evolutionary Algorithms for Stochastic Arc Routing Problems. In: Raidl, G.R., Cagnoni, S., Branke, J., Corne, D.W., Drechsler, R., Jin, Y., Johnson, C.G., Machado, P., Marchiori, E., Rothlauf, F., Smith, G.D., Squillero, G. (eds.) *EvoWorkshops 2004*. LNCS, vol. 3005, pp. 501–512. Springer, Heidelberg (2004)
- [9] Moscato, P., Cotta, C.: A Gentle Introduction to Memetic Algorithms. In: *Handbook of Metaheuristics*, pp. 105–144 (2003)
- [10] Fleury, G., Lacomme, P., Prins, C., Ramdane Cherif, W.: Improving Robustness of Solutions to Arc Routing Problem. *Journal of the Operational Research Society* 56, 526–538 (2006)
- [11] Mourao, M.C., Amado, L.: Heuristic Method for a Mixed Capacitated Arc Routing. *European Journal of Operational Research* 160(1), 139–153 (2005)
- [12] Mourao, M.C., Almeida, M.T.: Lower-bounding and Heuristic Methods for a Refuse Collection Vehicle Routing Problem. *European Journal of Operational Research* 121(2), 420–434 (2000)
- [13] Graham, R.L., Hell, P.: On the History of the Minimum Spanning Tree Problem. *IEEE Ann. Hist. Comput.* 7(1), 43–57 (1985)
- [14] <http://www.vv.es/belenguer/mcarp>

Impact of Indoor Location Information Reliability on Users' Trust of an Indoor Positioning System

Ting Wei and Scott Bell

Geography and Planning, University of Saskatchewan, 117 Science Pl, Saskatoon, SK
{Ting.Weil,Scott.Bell}@usask.ca

Abstract. Indoor positioning systems are used as a supplement to GPS where the satellite based technology does not work appropriately. However, positioning accuracy varies among techniques and algorithms used; system performance is also affected by local traffic and environmental structure. A relatively little studied topic is the effect of positioning variance on a user's opinion or trust of such systems (GPS as well, for that matter). An experiment was designed to examine how trust changes with positioning accuracy and whether trust can be built and maintained over time despite changes in positioning accuracy. We used a simulated version of our existing indoor positioning system to present groups of users with a series of positioning results with varying accuracy. Positions fell into one of three categories: 1. ACCURATE (<5 meters of error), 2. INACCURATE (>15 meters), and 3. WRONG BUILDING (outside current building). When a user experiences a series of accurate results first their trust of later inaccurate positioning is different from users who experience inaccurate locations first.

Keywords: Positioning systems, trust, individual differences.

1 Introduction

The Global Positioning System (GPS) provides accurate, reliable, and ubiquitous positioning in outdoor environments but unfortunately fails to provide reliable positioning indoors. As a result, several supplementary techniques have been used (Bluetooth, Cellular, wireless internet (WiFi), Radio Frequency ID (RFID), Ultra Wide Band (UWB), etc.) to provide positioning in settings where GPS does not function properly [1, 2]. Such systems can provide accurate locations, but all have characteristics that result in uncertainty (both between and within-system variation). In addition, improving accuracy requires one or more of the following: additional power, additional equipment/infrastructure, and/ or additional system latency [3], none of which can be increased infinitely or without additional cost. Under these relatively uncertain conditions a user's trust in positioning results may vary in conjunction with accuracy or might vary in a more complicated way related to their personal knowledge, experience, or the pattern of results they have experienced while using the system.

Trust is a relatively new concept to GIS, GPS, and the use of geographic information; however, it has been widely used for evaluating the usability of computer programs and related technology, such as our trust of web-based services [4-6] and collaborative computing systems [7]. Trust has been defined differently by

researchers depending on the entity to be trusted, the definition of the entity, and user characteristics [8]. Trust is an essential consideration for human computer interaction; research suggests there are many variables that affect user trust in computational settings [9]. To make matters more complicated, a user may initially trust a computational system only to distrust it later; it is also possible for a user to regain trust after initially distrusting a system [10]. While the reliability of a system might not immediately affect a user's trust, it seems safe to suggest that a user's trust of a system is dynamic, ongoing, and complicated [11]. For positioning systems, there lacks a clear definition of trust or a model that clarifies factors that can help establish user trust. In our first paper on this topic [12] we use the following positioning specific definition of trust: "a user's opinion of the positioning results which will affect their adoption and commitment to the system and their use of the information it provides." In addition, we suggest using four elements for modeling trust: Calculation method, Source data, User, and Graphic User Interface (GUI). Trust can be increased by improving perceived accuracy (from a user's perspective), which not only requires an accurate calculation with high quality data, but also a clear interface that communicates position as well as uncertainty (or accuracy) [12].

Research on in-car navigation systems indicates that accuracy affects users' opinions of system credibility as well as their attitudes to the car [13,14]. Interestingly, it was found that user trust increases with risk, that is to say, the user still trusts the system even as it puts them at risk [15]. This is related to the systems' role in a user's search for solutions to problems (such as being lost) and our increasing reliance on external solutions as problem complexity increases and our personal resources prove inadequate [16]. For indoor positioning systems, as mentioned before, the system is not always accurate and/or reliable. Inaccuracies caused by inadequate and inaccurate positioning source data, deficient techniques or algorithms, or an unclear GUI make it difficult to control or predict accuracy in real world settings. In addition, indoor spaces are generally considered qualitatively different from outdoor spaces [17]. Structural elements constrain navigation freedom, limit available decisions, and reduce visual extent. Systems currently available to support indoor navigation are just now beginning to incorporate complete and accurate floor plans; unfortunately, such systems are only available for a limited set of spaces and do not incorporate positioning information with the same accuracy or reliability as is available with GPS outdoors. These two characteristics of indoor navigation support (basemap data and positioning) result in much greater uncertainty in location information and highly unreliable positioning information. As a result, it is important to understand how user trust changes with system performance as well as develop models for indoor positioning systems to predict when results might be more uncertain. GPS is not immune to such trust issues, but we assume they are more infrequent; here we are most interested in emerging systems that may appear to function like GPS (accurate positioning with accurate and complete road network data). It is our strong opinion that indoor and outdoor systems are quite different in that indoor positioning systems are error prone, to the point that our research with various commercial systems (Google, SkyHook, and iOS) will place a user outside (beyond a buildings boundaries), when such a location is impossible. This result is highly egregious since almost all mobile devices that would provide such results include an Assisted-GPS chip and should be able to establish that the device is NOT

outdoors. However, most research treats trust as synonymous with accuracy or usability without considering the dynamic nature of system accuracy and user needs and their interaction with the surroundings of the individual. In addition, people do not think or respond identically under similar environmental conditions; for instance, personal navigation experience, individual differences, and spatial abilities might play an important role in the way trust changes during system use.

We designed an experiment to evaluate the impact of varying accuracy and reliability in indoor positioning on user trust. The simulated positioning system (embedded in the SaskEXP experimental design software and installed on an iPad) used for this experiment provided 10 priming positioning results at a specific level of accuracy (ACCURATE, INACCURATE, and WRONG BUILDING, see below for details) before a larger random series of positioning results from the same categories. Participants were given an iPad running the simulated positioning system that included the presentations of positioning results at pre-selected sites in the experimental area. At each point, participants were asked to rate their confidence in the positioning results. Our expectation (hypothesis) was that starting with accurate results would cause higher trust ratings for inaccurate results presented later in the experiment. This hypothesis is based on the premise that a set of consistently accurate results presented in series would communicate to the user that the system is reliable to the extent that the user would “overlook” (have less distrust for) inaccurate positioning results presented in a following series that included results across a range of accuracies. Furthermore, we anticipated that starting (priming) with inaccurate results would negatively affect trust; users would continue using the system but would rate all subsequent results less trustworthy. Therefore trust will be lower if initial location information is deemed untrustworthy.

2 Methods

2.1 Participants

54 students (27 males and 27 females) between the ages 19 to 32 (Mean: 23.56, SD: 3.04) participated in the experiment during the fall term of 2011. Students recruited from a geography course were given bonus credit, others were given an honorarium after the experiment. There were 18 participants in each of three experimental groups (described below), the number of males and females in each group were equal. All participants had some experience with the university campus used in the study.

2.2 Materials

As WiFi signals are dynamic (even when a sensor is stationary [18]), the positioning results calculated by our existing system (Saskatchewan Enhanced Positioning System (SaskEPS)) fluctuate over time. While accuracy is high (for one location the error will range from 4 to 7 meters, with the calculated location shifting slightly, as with GPS) using such results would mean each participant would be rating a slightly different positioning result. It is therefore impossible to completely control the positioning error of the presented location (result location). In this study, the positioning error was too important as an independent variable to be allowed to

fluctuate (each participant should experience the same amount of error for a trial they are all experiencing). Therefore, a simulated positioning system (installed on an iPad) was used instead of our existing positioning system during the experiment. From an interface perspective the simulated system functioned like our existing system. With a basemap including building shapes, hallway outlines, and roads on campus, the simulated system presented the calculated locations (pre-determined points) on top of the basemap in the form of a green dot with green lines indicating the outline of the hallway in our experimental area (see Fig. 1 in section 2.5). Participants were not aware that locations were pre-determined or simulated.

2.3 Experiment Data

The experiment was conducted on the second floors of three connected buildings on campus (Kirk Hall, Agriculture Building, and Engineering Building). 75 experimental points (visited by participants in real time) were randomly generated in ArcGIS using building hallways as a constraint. Not all 75 points were used for each participant, a subset of 10 (of one type) were used in a priming activity so 20 of the 75 would not be seen by each participant (see below). These 75 locations represented trials in one of three “location accuracy” categories: 1. ACCURATE Locations (25 locations): these locations represent places for which the system provides accurate location information (the presented points are within 5 meters of their actual position); 2. INACCURATE Locations (25 locations): these locations are inaccurate in absolute terms (positioning error is larger than 15 meters) but accurate in nominal/relative location (correct building); and 3. WRONG BUILDING Locations (25 points): these locations are inaccurate in nominal terms (outside the correct building’s boundaries). Either 10 ACCURATE, 10 INACCURATE, and 10 WRONG BUILDING locations were used as priming locations for three experimental groups respectively (a participant only visited one category of locations); the remaining locations in each category make up 45 post-priming location trials which were visited by all participants after priming. Based on the hallway area of the three buildings, 15 points were selected from Kirk Hall, 30 from the Agriculture Building, and 30 from the Engineering Building. Similarly, the number of ACCURATE locations, INACCURATE locations, and WRONG BUILDING locations for each building were based on the same ratio. Each experimental point was moved in order for it to fall within its designated accuracy category (this was accomplished by applying a buffer equivalent to the inaccuracy necessary (less than 5 m, more than 15 m, etc.). For WRONG BUILDING locations, each experimental point was moved outside the building outline using the actual location as a reference. Experimental locations (the positions presented to participants) were selected to be reasonable to participants (in the hallway or in a location outdoors where someone could stand, etc.)

2.4 Experimental Design

Participants were randomly assigned to one of three groups: ACCURATE priming, INACCURATE priming, or WRONG BUILDING priming. Each participant visited a total of 55 locations in series: 10 priming locations from a single accuracy category followed by 45 post-priming locations across the three accuracy categories. Group 1

began with 10 ACCURATE priming locations; groups 2 and 3 began with 10 INACCURATE priming locations and 10 WRONG BUILDING priming locations, respectively. A Latin Square design was used to establish visiting sequences of priming and post-priming locations within each building (to avoid moving back and forth among buildings), which is shown as Table 1 (“K” represents Kirk Hall, “A” represents the Agriculture Building, and “E” represents the Engineering Building). In order to improve data collection efficiency, the internal visiting sequence of both priming and post-priming locations for each building was based on a clockwise direction or counter-clockwise direction, this optimized movement between test locations and among experimental buildings.

Table 1. Building visiting sequences for each group

		Priming Locations			Post-priming Locations		
Group 1	a	K	A	E	E	A	K
	b	A	E	K	K	A	E
	c	E	K	A	A	E	K
Group 2	a	K	A	E	E	A	K
	b	A	E	K	K	A	E
	c	E	K	A	A	E	K
Group 3	a	K	A	E	E	A	K
	b	A	E	K	K	A	E
	c	E	K	A	A	E	K

Finally, minor modifications were made for the following reasons: 1. Locations near one another were presented with similar accuracy, as a user might expect from a real system; 2. The number of locations presented between two WRONG BUILDING locations was not constant as such a pattern might be evident to participants. At each point, participants were asked to rate their confidence in the positioning results (defined as “position trust”). “Positioning trust” illustrates how much they trust the position calculated at an individual location by the positioning system, which was evaluated on a 5-point likert scale (5=“Very high,” 4=“High,” 3=“Neutral,” 2=“Low,” and 1=“Very low”). In this way, each group allows for the examination of how accuracy at previously visited locations (accuracy of priming locations) affects subsequent trust evaluation (trust of post-priming locations). Furthermore, the overall body of data allows for an examination of how different levels of accuracy (ACCURATE, INACCURATE, WRONG BUILDING) affect trust.

2.5 Procedures

Before meeting each participant, the simulated SasKEPS was set to display positioning results in a sequence according to one of nine sub groups above (including building visiting sequence). The experiment was conducted on the University of Saskatchewan campus; each participant was engaged for approximately 100 minutes. After signing the consent form participants were given brief instructions about the

experiment. Before visiting the experimental sites participants were given a short training session focusing on how the interface and program works. They were taken to all 55 experimental sites (every participant stood at the same location for a trial and the same positioning results were displayed); at each experimental site they located themselves using the simulated SaskEPS on an iPad running as part of SaskEXP and rated position trust for that location. Each time a position was calculated by the system the trust questions were presented and participants' answers were recorded (Fig. 1). Participants used their own judgment when providing "Position trust" ratings as they were told that there is no standard measurement for each level of "Position trust." As is clear in Fig. 1 participants also rated system trust, we do not report those results here as it appears participants treated each type of trust similarly. In addition, participants were encouraged to verbally provide their reasons for each trust rating (these comments were recorded by the researcher). As the system was not calculating location in real time participants were informed to initiate positioning only when they were at the designated experimental site.

Position Trust	System Trust
<input type="radio"/> Very High	<input type="radio"/> Very High
<input type="radio"/> High	<input checked="" type="radio"/> High
<input checked="" type="radio"/> Neutral	<input type="radio"/> Neutral
<input type="radio"/> Low	<input type="radio"/> Low
<input type="radio"/> Very Low	<input type="radio"/> Very Low

Fig. 1. Data recording interface of the simulated SaskEPS

3 Results

The ordinal data from position trust ratings were not normally distributed, violating an essential assumption in our preferred test, Multivariate Analysis of Variance (MANOVA) (we attempted several transformations to normalize the data). As a result we used the Kruskal-Wallis *H* test [19] to examine the trust variation in different accuracy categories (ACCURATE, INACCURATE, and WRONG BUILDING) regarding the 45 post-priming locations and the impact of priming location accuracy (starting with ACCURATE (G1), INACCURATE (G2), WRONG BUILDING priming locations (G3)) on users’ trust of ACCURATE, INACCURATE, and WRONG BUILDING locations in the 45 locations that followed priming. The Mann-Whitney *U* Test [19] was used for paired comparisons if the main effect of accuracy category was significant. Bonferroni corrections [20] were applied to adjust the pre-chosen significance level ($\alpha=0.05/3=0.0167$) to make the significance test more stringent.

3.1 Overall Results

Table 2 provides descriptive statistics for position trust for post-priming locations, “A” represents position trust of ACCURATE locations, “IA” represents position trust of INACCURATE locations, and “WB” represents position trust of WRONG BUILDING locations. In general and as expected, ACCURATE locations generate higher trust than INACCURATE locations and WRONG BUILDING locations are the least trusted, which can be concluded from the Kruskal-Wallis *H* test results ($H = 36.054$, $df = 2$, $p < 0.001$) followed by paired comparisons using the Mann-Whitney *U* Test ($p < 0.001$ for each pair). This is also true when applied separately to individual groups (p values of both tests are less than 0.001). In addition, users’ opinions of INACCURATE locations show more variance than that of ACCURATE locations and WRONG BUILDING locations (evident from standard deviation and between group variance).

Table 2. Descriptive statistics for Position Trust (18 participants in each group, which includes 15 ACCURATE, 15 INACCURATE, and 15 WRONG BUILDING locations after priming locations)

Trust	Accurate (A)			Inaccurate (IA)			Wrong Building (WB)			
	Mean (SD)	Min	Max	Mean (SD)	Min	Max	Mean (SD)	Min	Max	
Overall	4.46 (0.30)	3.55	4.83	2.21 (0.58)	1.11	3.72	1.57 (0.41)	1.05	2.67	
Group	G1	4.22 (0.09)	3.55	4.72	1.83 (0.11)	1.11	2.78	1.30 (0.05)	1.05	1.55
	G2	4.53 (0.06)	3.94	4.83	2.04 (0.10)	1.55	2.67	1.46 (0.07)	1.05	2.17
	G3	4.63 (0.04)	4.33	4.78	2.77 (0.13)	2.05	3.72	1.94 (0.11)	1.33	2.67

3.2 Trust by Group

The next step was to examine the role that priming effect plays in trust. Recall that we expected ACCURATE priming locations would increase subsequent trust. In fact, the opposite is true. The Kruskal-Wallis results indicate that for Group1, who were primed with 10 ACCURATE positioning results in the beginning, position trust ratings for all three types of locations (ACCURATE, INACCURATE, and WRONG BUILDING locations) are significantly different (lower) than ratings by participants who were primed with 10 INACCURATE or 10 WRONG BUILDING locations (comparisons among groups for post-priming ACCURATE locations: $H = 15.025$, $df = 2$, $p = 0.001$; for post-priming INACCURATE locations: $H = 19.617$, $df = 2$, $p < 0.001$; and for post-priming WRONG BUILDING locations: $H = 18.328$, $df = 2$, $p < 0.001$). Mean values of trust of three types of locations across groups are shown in Fig. 2. The overall pattern indicates that users' trust of the three location types is similar across groups. However, priming with WRONG BUILDING locations (Group 3) has a stronger effect on trust than INACCURATE priming (Group 2).

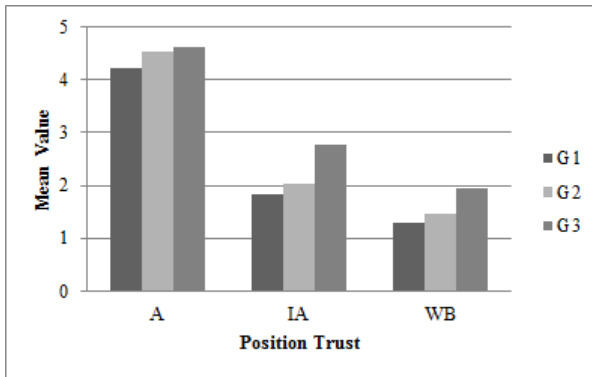


Fig. 2. Mean values of position trust of three location types (A: position trust of ACCURATE locations, IA: position trust of INACCURATE locations; WB: position trust of WRONG BUILDING locations; Groups, G1: ACCURATE priming, G2: INACCURATE priming, G3: WRONG BUILDING priming)

The pair-wised comparisons between groups for position trust of each type of location were examined using the Mann-Whitney U Test. Calculated p values can be found in Table 3 (* indicates significant at 0.0167 level). For ACCURATE locations users have more trust if they start with WRONG BUILDING locations and INACCURATE locations than when starting with ACCURATE locations. Regarding INACCURATE and WRONG BUILDING locations, starting with WRONG BUILDING locations results in higher trust compared to groups that start with ACCURATE and INACCURATE locations. In addition, priming with WRONG BUILDING locations produces the highest ratings of trust.

Table 3. Calculated *p* values from the Mann-Whitney *U* Test

	A			IA			WB		
	G1	G2	G3	G1	G2	G3	G1	G2	G3
G1	-	0.005*	<0.001*	-	0.161	<0.001*	-	0.161	<0.001*
G2	-	-	0.187	-	-	<0.001*	-	-	0.002*
G3	-	-	-	-	-	-	-	-	-

Upon closer inspection we discovered interesting patterns among types of locations and groups; these results are presented as a summary of significant differences in means of trust ratings in table 4. For ACCURATE and INACCURATE locations, the significant trust increase occurs in groups when priming locations are in less accurate categories. For WRONG BUILDING locations, as they are in the least accurate category, the significant trust increase occurs in groups when priming locations are in the equivalent category (Group 3: WRONG BUILDING priming). It can be concluded that experiencing less accuracy first, especially WRONG BUILDING locations (strongest effect), has a positive impact on users’ trust of positioning results that follow; conversely, when starting with ACCURATE priming locations trust in all following locations declines.

Table 4. Significant scenarios of paired comparisons for trust of three types of locations across groups

Mean value comparison (significant scenarios)	
A	Starting with WB (G3) and IA (G2) > Starting with A (G1)
IA	Starting with WB (G3) > Starting with IA (G2) and A (G1)
WB	Starting with WB (G3) > Starting with IA (G2) and A (G1)

4 Discussion

4.1 Positioning Accuracy

For all three groups user trust is substantially higher for ACCURATE locations. Users trust ACCURATE locations most followed by INACCURATE locations; WRONG BUILDING locations are the least trusted. These results have implications for anyone developing a positioning system. For an individual location, it can be concluded that user trust is higher when accuracy is higher; when accuracy is lower the designer should take measures to keep the user engaged and not lose them to distrust. In this experiment users had little option to stop using the positioning system (they were free to discontinue the experiment, but as long as they were in the experiment they had to provide a trust rating). This is not true of positioning system use in reality; people are more likely to put the system away or turn it off than suffer through inaccurate results. In the case of WRONG BUILDING locations, we recommend the incorporation of a constraint that wouldn’t let a position be displayed beyond a building’s footprint.

Since most devices that would be running such a system (for instance, SasKEPS runs on Android and Windows operating systems) engaging the GPS sensor is the best way to determine if the device is outdoors (if the system sees several GPS signals, some of relatively high strength, there is a high likelihood that the device is outdoors, otherwise it is NOT). According to our results, although there was variability in ratings of WRONG BUILDING locations, this can be partially explained by the relatively lower absolute error in some of these cases (displayed location was outside, but relatively close to the user's actual location). However, from an overall point of view, it is a universal law that users trust locations within building outlines more than those outside building outlines.

4.2 Priming Effect

Interestingly, our hypothesis regarding the impact of priming with locations in different types of accuracy was refuted by our results. As mentioned in the results section, priming with inaccurate locations, especially WRONG BUILDING locations, can increase trust of later positioning results. Users tend to have higher expectations when they initially see a series of accurate locations. On the contrary, seeing inaccurate locations first decreases users' initial expectations, making subsequent locations seem more trustworthy. One way of thinking of this pattern is that seeing consistently accurate results initially sets the bar high, subsequent inaccurate locations don't live up to expectations and are therefore rated as far less trustworthy. This effect extends to locations that are as accurate as the priming locations.

In a real world setting, positioning and associated accuracy change continuously. When applying the above patterns, it seems unwise for a positioning system to be initially accurate and then lose accuracy. Specifically, if a positioning system functions well for a period of time users will expect to have at least equivalent accuracy later. When system accuracy is low for a specific and possibly knowable reason (e.g. out of system range, not enough available signals), user trust will decline; this negative impact will likely extend to subsequently accurate results as well. To avoid such an outcome, indoor positioning systems should target consistent accuracy. While our results suggest that trust can be recovered from inaccurate results, we don't consider the resilience of trust as a reason to expect users will continue to use an inaccurate system. In fact, we advocate for using the GUI to clearly communicate to users as much about the positioning process to reduce the inflation of expectations. It is possible that most users will continue with a system with periodic and one-time inaccuracy, but each subsequent inaccurate location increases the likelihood that a user will stop using the system (and not return). As a result, the system must provide information explaining inaccuracy; we believe such communication offers a good opportunity for users to regain trust before abandoning the system.

5 Conclusion

Indoor positioning systems provide an alternative solution for navigating indoor environments where GPS does not function. Such systems are generally less accurate

and reliable than their GPS counterparts. Therefore, understanding how users interact with changes in accuracy is important for achieving a more usable positioning system design. The existing literature evaluates user trust of a positioning system only based on one-time or average positioning accuracy and does not take into account the dynamic interaction among the user, device/system, and trust. Results from our experiment provide an initial framework to understand the relationship between positioning accuracy and human trust, particularly how initial accuracy can affect users' trust of subsequent positioning results. This pattern provides support for understanding the complex nature of trust and geospatial data, processes, and representations. Trust is not simply shaped by a single or periodic assessment of accuracy, but maintained by consistent positioning accuracy throughout the whole positioning process. Specifically, users' trust of positioning results changes with the system performance over time, which is affected by their previous experience and current accuracy. As inaccuracy can occur indoors and outdoors and not only at already known locations, it is important to maintain user trust especially when the positioning information is less reliable. This research will help both designers of positioning systems and researchers alike to understand the nature of trust change and improve the efficiency of user-system interaction during periods of suspect accuracy. Future work should be built on our model of trust as well as develop increasing nuance in the user system trust relationship.

Acknowledgements. The authors would like to thank the University of Saskatchewan Information Technology Services and Facilities Management Division for their willingness to share the campus map data. The authors are indebted to the Canadian National Centers for Excellence, GEOIDE for funding as part of both Phase IV (PIV - 03) and Short Strategic Industrial Initiative (SSII - 107). The team would also like to acknowledge space and equipment provided by the Spatial Analysis for Innovation in Health Research (SAFIHR) Lab with funding from Canadian Foundation for Innovation (CFI).

References

1. Georgy, J., Noureldin, A., Korenberg, M.J., Bayoumi, M.M.: Low-Cost Three-Dimensional Navigation Solution for RISS/GPS Integration Using Mixture Particle Filter. *IEEE Transactions on Vehicular Technology* 59(2), 599–615 (2010)
2. Georgy, J., Noureldin, A., Korenberg, M.J., Bayoumi, M.M.: Modeling the Stochastic Drift of a MEMS-Based Gyroscope in Gyro/Odometer/GPS Integrated Navigation. *IEEE Transactions on Intelligent Transportation Systems* 11(4), 856–872 (2010)
3. Beal, J.: Contextual geolocation, a specialized application for improving indoor location awareness in wireless local area networks. In: *The 36th Annual Midwest Instruction and Computing Symposium (MICS 2003)*, Duluth, Minnesota, pp. 1–17 (2003)
4. Roy, M.C., Dewit, O., Aubert, B.A.: The impact of interface usability on trust in web retailers. *Internet Research* 11(5), 388–398 (2001)
5. Flavián, C., Guinalú, M., Gurrea, R.: The role played by perceived usability, satisfaction and consumer trust on website loyalty. *Information & Management* 43(1), 1–14 (2006)

6. Artz, D., Gil, Y.: A survey of trust in computer science and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web* 5(2), 58–71 (2007)
7. Thirunarayan, K., Anantharam, P., Henson, C.A., Sheth, A.P.: Some trust issues in social networks and sensor networks. In: *International Symposium on Collaborative Technologies and Systems (CTS)*, pp. 573–580. IEEE, New York (2010)
8. Lee, J.D., See, K.A.: Trust in automation: Designing for appropriate reliance. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 46(1), 50 (2004)
9. Fogg, B., Tseng, H.: The elements of computer credibility. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: The CHI is the Limit*, pp. 80–87. ACM, New York (1999)
10. Tseng, S., Fogg, B.: Credibility and computing technology. *Communications of the ACM* 42(5), 39–44 (1999)
11. Lee, J.D., See, K.A.: Trust in computer technology and the implications for design and evaluation. *Etiquette for Human-Computer Work: Technical Report FS-02-02*, pp. 20–25 (2002)
12. Bell, S., Wei, T., Jung, W.R., Scott, A.: A conceptual model of trust for indoor positioning systems. In: *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*, pp. 7–14. ACM, New York (2011)
13. Jonsson, I., Nass, C., Harris, H.: How accurate must an in-car information system be?: consequences of accurate and inaccurate information in cars. In: *26th Annual SIGCHI Conference on Human Factors in Computing Systems*, pp. 1665–1674. ACM, New York (2008)
14. Jonsson, I., Nass, C., Harris, H., Takayama, L.: Got Info? Examining the Consequences of Inaccurate Information Systems. In: *3rd International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, pp. 409–415, Public Policy Center, University of Iowa, Iowa City (2005)
15. Perkins, L.A., Miller, J.E., Hashemi, A., Burns, G.: Designing for Human-Centered Systems: Situational Risk as a Factor of Trust in Automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 54(25), 2130–2134 (2010)
16. Ishikawa, T., Fujiwara, H., Imai, O., Okabe, A.: Wayfinding with a GPS-based mobile navigation system: A comparison with maps and direct experience. *Journal of Environmental Psychology* 28(1), 74–82 (2008)
17. Li, K.-J.: Indoor Space: A New Notion of Space. In: Bertolotto, M., Ray, C., Li, X. (eds.) *W2GIS 2008. LNCS*, vol. 5373, pp. 1–3. Springer, Heidelberg (2008)
18. Xiang, Z., Song, S., Chen, J., Wang, H., Huang, J., Gao, X.: A wireless LAN-based indoor positioning technology. *IBM Journal of Research and Development* 48(5.6), 617–626 (2004)
19. Heiman, G.W.: *Basic statistics for the behavioral sciences*. Wadsworth Pub. Co., Belmont (2010)
20. Elzinga, C.L.: *Monitoring plant and animal populations*. Wiley-Blackwell, USA (2001)

Ontology for the Engineering of Geospatial Systems

Nancy Wiegand

University of Wisconsin-Madison, Madison, Wisconsin
wiegand@cs.wisc.edu

Abstract. In this paper, a metamodel ontology is introduced to describe a domain of data components for geospatial data and query systems. The ontology satisfies the need to model the more complex environment that occurs within a geospatial system. For example, contrary to typical databases, geospatial data have additional metadata files describing the actual data. Also, a geospatial system may have domain ontologies in addition to semantic mappings. Currently, user knowledge is required to know the relationships between all data components (data, metadata, ontologies, mappings, etc.). Contrary to that, we propose a system ontology over which automatic inferencing can be done to determine relationships and meanings among data components. This work fits into the vision of the Semantic Web and interlinked data and knowledge networks and applies these notions to a metamodel for a data system.

Keywords: Engineering design and management of geospatial information systems, ontology, semantics, metadata, inference, metamodel, query.

1 Introduction

A geospatial data and query system is already inherently more complex than other data systems because of the associated geospatial metadata. With the addition of domain ontologies and mappings, geospatial, as well as other information systems, are becoming more complex. Many types of data components now need to be managed. This paper gives an example of how to model and manage related data components in geospatial systems using an ontology for the software engineering design of the system itself.

The architecture in which the system level ontology could be used can vary. Initially, this could be thought of as being within a Database Management System (DBMS), and this paper explores that type of system to explain the ideas. But, the design and deployment could be used in geospatial portals, federated systems, a cyberinfrastructure, or a cloud architecture.

Geospatial data have associated separate metadata files e.g., [4, 6], possibly more than one per data set. In the geospatial realm, the querying of metadata files tends to be mainly used in searching for the associated data file. For example, geospatial portals, such as Geospatial One-Stop (GOS) [9] (now part of Data.gov), accept user keywords to search metadata to locate a data set, rather than searching the data files themselves [24].

In addition to metadata, geospatial data systems may now have ontologies and mapping files to manage heterogeneous attributes and values. Although resolving heterogeneous schemas is a long-studied issue in the database community with Local As View or Global As View as possible solutions, other ways of attempting to deal with the problem now use formal semantic technologies. That is, people create OWL [23] ontologies and then use reasoners to infer subclass or superclass relationships or to disambiguate terms. Ontologies, however, create extra files to be managed in a data system. In addition, there may also be multiple files of pre-set ontology mappings to be used for lookup at query time for query re-writing. The ontologies and mapping files either need to be stored in a data system or available remotely. To handle the growing need to manage and access different types of files, a data and query system needs a more complex engineering model. There should be a method in an enhanced information system to inherently associate files with each other. Fig. 1 shows different types of possible files and the relationships between them. For example, ontologies may describe one or more data files and/or one or more metadata files and/or individual attributes within data or metadata files.

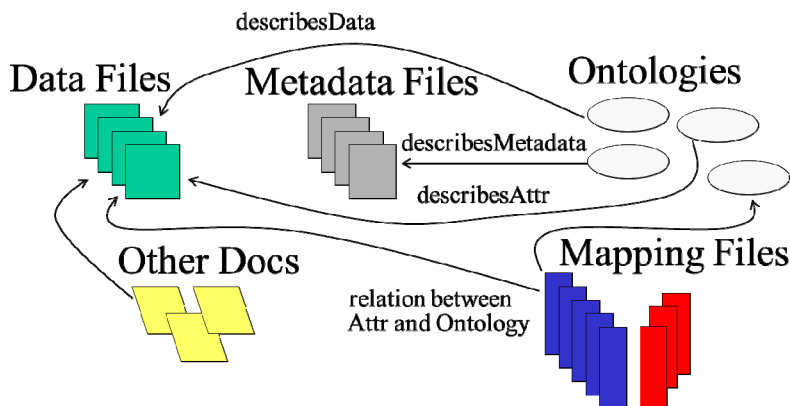


Fig. 1. Different types of files in a data system and their relationships

A portal, DBMS, or other type of information system could now contain or need access to data files, metadata files, multiple ontologies for various subdomains or attributes, semantic mapping files, and other associated data. Portals, if they include semantic support, tend to be hard-coded with pre-set ontology files for a particular application. And, although some current DBMSs now have support for ontologies for use in query expansion, they still are not able to model the association or relationship between data and metadata, between data and ontologies, between metadata and ontologies, between attributes and ontologies, nor between attributes in either data or metadata and their ontology mapping files. Full management of all types of files, along with the associations between them, should be known and handled at a high level of abstraction. To solve this problem, we propose a system ontology to describe the inter-relationships among data components.

An ontology is a formal representation of the concepts, terms, and relationships in a domain. Normally, a domain might be something such as hydrography, for example, and an ontology would describe terms having to do with water or navigable waters. Here, however, the domain consists of types of files that might occur in a system. We use the term *data component* (such as data files, metadata files, ontologies, mappings, schemas, and other related components) to represent a type of data stored and managed in a data system. The data component ontology represents the engineering design of the system architecture regarding the ‘data’.

The contributions of this paper are:

- to motivate the need to handle more complex modeling within geospatial data and query systems,
- to design a system level conceptual model that is extensible,
- and to formally represent the model and show its use as to how it frees the user in posing queries and frees application programmers from having to know or write specific code to handle details of data components and their relationships.

Section 2 gives related work. Section 3 presents a geospatial data example, and Section 4 presents a potential geospatial data component ontology. An instantiation of the ontology is queried in Section 5. Section 6 gives a discussion and conclusions.

2 Related Work

This work fits into the overall vision of a Semantic Web with linked data [1, 14]. In that vision, information is linked through relationships forming a network of interconnected pieces of knowledge. Inference is done through links to discover information not directly available. Here, we propose to use this capability within a data system itself to link data components to alleviate users and application programmers from having to know (or hard code) the names and relationships of files.

Geographers have related ontologies to geospatial systems to be able to semantically describe attributes and to help resolve semantic discrepancies [e.g., 5, 8, 12, 21, 29, 30, and 31]. In this work, however, we use an ontology to define the space of data components in an information system.

To help build a Semantic Web with linked data, geographic linked data has been put on the Web by the U.K. Ordnance Survey as reported in [10]. The U.S. government has an open and linked data initiative as part of Data.gov [3] with a large collection of RDF data sets.

Ontologies have been compared to DBMS schemas [27], which have a closed versus open world assumption. However, we use an ontology to organize various data components, of which a schema could be just one component. Although some ontology support has been added to DBMSs, an overarching model of components has not been added. Using ontologies in DBMS queries was introduced in [25] with further work of theirs on semantics in [28]. Oracle Spatial 11g [2, 17] and DB2 [15] have included semantic support. However, the user must know and specify the name of the relevant ontology.

Modularizing subsystems of DBMSs was proposed in [22] because of new types of uses for database technology. Along these lines, a DataSpace Support Platform (DSSP) [11] was introduced to accommodate data management for loosely connected, but related, data. Managing relationships between the sources, however, was not a focus of a DSSP. Object-Based Data Access (OBDA) systems are being developed that use ontologies to mediate access to data. Abstraction to this methodology is found in [20]. Their abstractions, however, are not similar; they are not modeling all data components in the system.

3 Geospatial Example

Geospatial data motivated this work because of the additional geospatial metadata files that describe data sources. A metadata file has its own schema which may be that of the Federal Geographic Data Committee (FGDC) metadata format [6]. The Theme_Keyword of FGDC metadata is often used for matches in keyword searching for data sources in geospatial portals such as Geospatial One-Stop [9].

For an example, we propose the data component instances shown in Fig. 2 and Table 1. The ‘hydroDaneCtyWI’ dataset contains hydrography data for Dane County in Wisconsin. It is described by the ‘FGDC_metadata1’ metadata file. There is another metadata file describing it in Dublin Core format called ‘DC_metadata1’. The ‘hydroECCTyWI’ has hydrography data for Eau Claire County, Wisconsin. Its FGDC metadata file is called ‘FGDC_metadata2’. The hydrography data sets have an associated ontology called ‘HydroOntology’, such as [13]. Terms from this ontology can also be associated with attributes in addition to the general data domain. Here, the FGDC_Theme_Keyword attribute is specifically associated with HydroOntology. Finally, the ‘hydroDaneCtyWI’ data set has a ‘flow’ attribute, ‘hydroDaneCtyWI_flow’, shown in the left bottom corner, to measure water flow, and there is an ontology for measurements [e.g., 16] that can be used to further describe flow. Table 1 uses the classes modeled in Fig. 3 to show the types of the instances from Fig. 2.

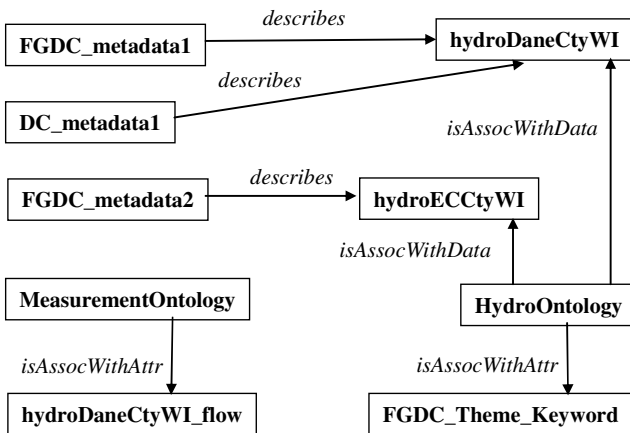


Fig. 2. Diagram of data component instances and their relationships

Table 1. Example data component instances and their classes

Instance	Class
FGDC_metadata1	FGDC_Metadata
DC_metadata1	DublinCore_Metadata
FGDC_metadata2	FGDC_Metadata
hydroDaneCtyWI	DataSet
hydroECCtyWI	DataSet
FGDC_Theme_Keyword	Attribute
hydroDaneCtyWI_flow	Attribute
HydroOntology	Ontology
MeasurementOntology	Ontology

4 Data Component Ontology

We modeled a basic template for a geospatial data component ontology based on our experiences with geospatial data (Fig. 3). The model is extensible, however. We used semantic technologies to create and use this model, i.e., Protégé [18] and SPARQL [19]. Although possible, it would be cumbersome to do this modeling using additional relational schema tables in a DBMS, for example, whereas OWL [23] is well suited for this. Using OWL enables a reasoner to discover components, their types, and relationships. Because Oracle, for example, already has a reasoner as part of its ontology system and a version of SPARQL for querying, adding a data component ontology to a DBMS could be done, as it could also be done in other architectures.

Fig. 3 has a class hierarchy showing possible components of a data system. The example given here models metadata, data sets, attributes, auxiliary documents, ontologies, and ontology mappings as types of data components. Here, the class ‘ontology’ represents a collection of data components that are each an ontology. Only base components are modeled in this example, but other components and relationships could be added or modified by a DBA. Here, we do not model schemas, data dictionaries, or other DBMS components. Instead, we focus on additional components currently not modeled.

In Fig. 3, metadata files and data sets are declared as subclasses of Data because they are similar in that they both have their own schemas, contain attributes, and may be associated with ontologies. As an example template for geospatial applications, two types of metadata are shown, FGDC and Dublin Core.

Attributes are declared as their own class because they may be independently associated with ontologies, separately from the data set in which they reside. There can be many ontologies associated with data and attributes. Attributes may also have ontology mappings to resolve heterogeneous values between data sources. Although not modeled here, schema mappings between heterogeneous schemas would be another type of ontology mapping.

```

Class DataComponent
  Class Data
    Class Metadata
      Class FGDC_Metadata
      Class DublinCore_Metadata
    Class DataSet
  Class Attribute
  Class AuxiliaryDocument (e.g., document,
    spreadsheet)
  Class Ontology
  Class OntologyMapping
    
```

Fig. 3. A data component ontology, as a model for ontology template classes for data components. Indentation shows subclassing.

Object properties between classes are shown in Table 2. The model is graphically represented in Fig. 4.

Table 2. Relationships between data components in an ontology

Domain	Object Property	Cardinality	Range
Metadata	describes	one	Dataset
DataSet	isDescribedBy	0 or more	Metadata
Data	hasDataOntology	0 or more	Ontology
Ontology	isAssocWithData	0 or more	Data
Attribute	hasAttrOntology	0 or more	Ontology
Ontology	isAssocWithAttr	0 or more	Attribute

5 Searching and Querying over the Data Component Ontology

This section shows the advantages of having a data component ontology and some of the reasoning possible over it. Advantages include the application system being able to automatically determine the data components and their relationships without the user having to know them. Files of a certain type can be inferred from the ontology class hierarchy.

To run these queries, instances for our geospatial example were added to the template ontology, resulting in an **instantiation of the model**. We used Protégé to develop the ontology template, create an instance of the template, and generate an OWL file. Then to query this file, we used Twinkle [26], a SPARQL query tool with a GUI interface that wraps the ARQ SPARQL query engine. The OWL file contains the data shown in Fig. 2 and Table 1.

This section is divided into examples of first using the data component ontology to find data sets associated with a particular instance ontology. The data component ontology is then used for searching (versus querying, meaning that searching is being

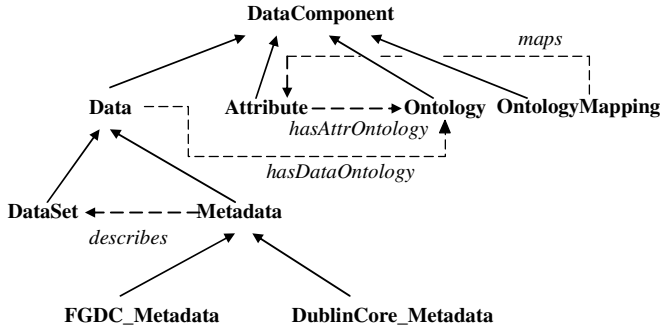


Fig. 4. Ontology template showing object properties with dashed arrows. Solid arrows show subclassing.

done over metadata files as distinct from the data files themselves). The data component ontology is then used for query expansion while searching and finally used to query the data sources themselves when limited information is available.

5.1 Querying the Data Component Ontology - Examples

The OWL data component ontology with instances can be directly queried for information. We start with some straightforward examples. (In SPARQL, the SELECT clause states what is to be returned, and the WHERE clause contains one or more triple patterns to be matched. The stated PREFIX ‘hydro’ for the OWL ontology used in this paper is shown once below, then assumed for all queries and not repeated.

Q1 and Q2 show how a reasoner as part of a query processor could automatically find information needed to generate queries by inferring over the instantiated metamodel. Q1 finds the data set(s) associated with the hydrology ontology by inferencing over an object property. Because ‘isAssocWithData’ has a range of type DataSet, ‘?d’ will return data sets. The hydroDaneCtyWI data set is returned. Q2 automatically finds the needed ontology to consult when querying over the Theme metadata element.

Q1. Find the data set(s) associated with the HydroOntology

PREFIX

```
hydro:<http://www.semanticweb.org/ontologies/2010/3/9/Ontology1270863566125.
owl#>
```

```
SELECT ?d
```

```
WHERE { hydro:HydroOntology hydro:isAssocWithData ?d }
```

Answer:

```
http://www.semanticweb.org/ontologies/2010/3/9/Ontology1270863566125.owl#hy
droDaneCtyWI
```

Q2. With what ontology, if any, is the attribute FGDC_Theme_Keyword associated?

```
SELECT ?o
WHERE { hydro:FGDC_Theme_Keyword
        hydro:hasAttrOntology ?o }
```

Answer:

<http://www.semanticweb.org/ontologies/2010/3/9/Ontology1270863566125.owl#>>
HydroOntology

5.2 Search

In this section we show the advantages of having a data component ontology when searching for geospatial data sets in a more complex environment, such as shown in Fig. 1. Searching for geospatial data has always been a separate type of querying compared to querying the data sources themselves because a geospatial search query typically queries the metadata (and not the data itself). In a spatial data infrastructure, such as Geospatial One-Stop [9] for example, a prior-created combined metadata database is searched when the user types in a keyword and location to find data sets, e.g., theme keyword = 'river'. Metadata records matching the conditions have direct or indirect access information to their associated data sets, such as a URL allowing the user to download the data.

An enhanced information system contains many types of 'data'. However, we only want to execute the search query over the metadata files. We do not expect the user to know the names of the metadata files. The challenge is to automatically determine which files are metadata files (Q3), over which the search can then be done for the user keyword and location (Q4). To start, we show Q3 that uses inference over the OWL instance ontology to find all the metadata files. Q3 is then part of Q4.

Q3. Find files that are of type FGDC Metadata

```
SELECT ?m
WHERE { ?m <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
        hydro:FGDC_Metadata }
```

Answer:

http://www.semanticweb.org/ontologies/2010/3/9/Ontology1270863566125.owl#FGDC_Metadata1
http://www.semanticweb.org/ontologies/2010/3/9/Ontology1270863566125.owl#FGDC_Metadata2

Query 3 selects files of an RDF 'type' FGDC_Metadata. As can be seen in the answer above and in Table 1, two files result, FGDC_Metadata1 and FGDC_Metadata2. This query illustrates that it is possible to find all files of a certain type without having to specify the names of the files. **Contrary to this, in a Relational DBMS SQL query, there is typically no way to determine files of a certain type.**

Q4. User search query: Find the URLs of Data Sets with Theme Keyword = ‘river’

Finding files of type metadata as in Q3 is the first step in the geospatial search query to find all data sets having to do with rivers. The full query below returns the URLs of the data sets for which their metadata has theme keyword = ‘river’.

To do this query, we need the ‘hasDataSetURL’ and ‘hasThemeKeyword’ OWL object properties for the FGDC_Metadata class, which we have not shown so far, along with additional instances for the data set URLs, theme keywords for individual metadata files, and other values in the instantiation. Such additions were made to the data component ontology for this application.

```
SELECT ?datasetURL
WHERE { ?m a hydro:FGDC_Metadata .
        ?m hydro:hasDataSetURL ?datasetURL .
        ?m hydro:hasThemeKeyword ?tk .
        ?tk hydro:hasValue "river"
      }
```

Answer:

```
...FGDC_DataSetURL1
```

Executing the above query involves reasoning over the class ontology as well as more typical query processing. As in Q3, finding metadata to search over is done by evaluating the first triple of the WHERE clause (‘a’ is a shortcut for ‘type’). The second triple specifies the data set URL stored in the metadata file. The third triple specifies the theme keyword of the metadata file, and the fourth triple matches the theme keyword having the value “river”.

Q5. Expand on Q4 to find data sets when the data set URL is not available in the metadata files. This uses the relationship between data and their metadata files

This query shows another example of the use and value of inference. Data.gov is a new portal holding general government data and subsumes Geospatial One-Stop. Because in actuality many metadata files do not contain the URL for the data sources they describe and because data.gov does not have other viewing services, work is being done to update data.gov metadata files to include the URL of the data set to enable download [7]. In Q5 we now show that the URL would not be needed in our approach because the relationship between metadata and data can be inferred in the instance ontology. That is, the data set described by the metadata file can be found using the ‘describes’ object property. With that relationship available, it is not necessary to have the data set URL available in the metadata file.

Q5. Find data sets where the metadata theme keyword = ‘river’ using the inferred association between a metadata file and its data set

```
SELECT ?d
WHERE { ?m a hydro:FGDC_Metadata .
        ?m hydro:describes ?d .
        ?m hydro:hasThemeKeyword ?tk .
        ?tk hydro:hasValue "river"
      }
```

Answer:

```
...FGDC_DataSetURL1
```

5.3 Search Using Query Expansion Compared to Oracle

The above queries do an exact match for the value of Theme_Keyword equal to 'river'. Contrary to this, an ontology-assisted search (query expansion) would also include terms that are related to the term river, e.g., stream, creek. In the geospatial context, ontology search was presented, for example, in [12].

To start this example, the syntax for semantic search using query expansion in Oracle Spatial 11g, a relational DBMS, is shown. Italics are used to indicate where the user has to supply precise names (Fig. 5).

```
SELECT DataSetURL
FROM a specified relational data table
WHERE SEM_RELATED (Theme_Keyword, 'rdfs:subClassOf', 'river',
    sem_models (a specified domain ontology)) = 1;
```

Fig. 5. Semantic Search in Oracle Spatial 11g [2]

An example for the query in Fig. 5 would be to find the data set URL from a metadata table, as found in geospatial portals, where not only does the theme keyword equal 'river' but a domain ontology is consulted for additional terms, here subclasses, such as 'stream' or 'creek'. Such a query is somewhat similar to Q4 in that the data set URL will be returned for tuples that have theme keyword equal to 'river', except here, any theme keyword term that is a subclass of the term 'river' as found in a domain ontology will also be returned.

Contrary to the Oracle syntax, the objective in this paper is to use data component ontology information to infer as much as possible so that the user does not need to know, for example, the existence or name of a domain ontology. For example, according to Fig. 2, the associated ontology for FGDC_Theme_Keyword is 'HydroOntology'. This can be inferred using the data component ontology instantiation so it does not need to be stated.

The following query builds on query Q4 to do query expansion. The new parts of the query are to automatically find the ontology associated with theme keyword and find subclass terms in that ontology. An additional prefix is needed to indicate where the hydrography ontology is located, which here is considered to be a taxonomy of terms.

Q6. Find data set URLs where the value of theme keyword is a subclass of 'river'

```
PREFIX
hydro:<http://www.semanticweb.org/ontologies/2010/3/9/Ontology1270863566125.owl#>
tax:http://www.taxonomy.org/etc.
SELECT ?datasetURL
WHERE { ?m a hydro:FGDC_Metadata .
    ?m hydro:hasDataSetURL ?datasetURL .
    ?m hydro:hasThemeKeyword ?tk .
    hydro:FGDC_Theme_Keyword hydro:hasAttrOntology ?o .
    ?o tax:hasTerm ?term .
    ?term tax:hasValue "river" .
    ?subterm rdfs:subClassOf ?term .
    ?tk hydro:hasValue ?subterm .
}
```

The first three triples in the above query are the same as Q4 and determine the data set URL and theme keyword in the metadata file. The fourth triple matches the domain ontology associated with each theme keyword. This subquery is the same query as shown in Q2. After this, we make the assumption that the domain ontology is organized as a taxonomy in which classes are the taxonomy terms. For example, water as a main class could have river and lake as subclasses. River then could have brook and creek as its subclasses, etc. We assume a prefix for a standard taxonomy is defined (i.e., ‘tax’ above), and the fifth triple defines terms in the taxonomy. The sixth triple determines the term with the value ‘river’, and the seventh triple finds subclasses of the term ‘river’. The final triple matches the value of each theme keyword to the subclasses of ‘river’.

The advantage of the above query is that the appropriate ontology for the theme keyword can be inferred rather than needing to be stated as in the ‘sem_models’ subclause in Fig. 5. In fact, a sem_models clause would not be needed. The user does not have to know the name of a domain ontology to do query expansion. This query was not executed because creating a domain ontology for hydrography is beyond the purpose of this paper, which is to show the value of having a data component ontology for an information system.

5.4 Query Over All Datasets Not Knowing Attribute Names (Query the Data Itself Versus Search the Metadata)

A user may want to search for the term ‘river’ within the data sets themselves instead of searching the theme keywords of metadata files. This is not done now in geospatial search systems but could greatly enhance finding geospatial data given increased data management capabilities.

In this situation, the attribute name, as well as the data set name, would be unknown. This is contrary to a normal SQL query such as `SELECT someAttribute FROM table WHERE attributeName = “river”` in which the names of the table and attributes need to be specified.

Here, we assume the term could occur within any attribute in any data set. Further even if the attribute name was known for one data set, independently created datasets could have heterogeneous attribute names even if the attributes are related. In this query, it does not matter. Using the data component ontology, the following query can be posed to return the data set(s) that contain the term ‘river’ somewhere, i.e., in some attribute.

Q7. Find data sets containing the term “river” somewhere, i.e., in any attribute

```
SELECT ?dataset
WHERE { ?dataset a hydro:DataSet .
        ?dataset hydro:hasAttribute ?attr .
        ?attr hydro:hasValue “river” }
```

Here, the query ranges over files of type DataSet. The ontology information allows the user to not know the names of the data files, just as in the prior queries where the

names of the metadata files were not needed. Further, the WHERE clause limits search to attributes that are part of data sets (versus attributes that are part of metadata or an ontology, for example). The names of the attributes are not needed. Attributes with the value 'river' are determined using the hasValue datatype property. Currently, such a query would not be processable in SQL, for example, although it could be done in data management systems using inverted lists.

Being able to execute such a query with limited knowledge is very powerful. This opens up new possibilities for the user to query over data without knowing identifiers for tables or attributes. This ability would be very useful within current DBMSs even without considering managing additional data component files.

6 Conclusions

With the increased interest in and need for geospatial data, more comprehensive data component management is needed in geospatial information systems. We applied Semantic Web technologies, e.g., OWL and SPARQL, to design a generic system level ontology to model the types of data components that may now be found in a full data system. The types of data relevant to an application have expanded beyond the typical tables in a Relational Database Management System, for example, such that an upper layer of meta-information is now needed to manage relationships between the components as part of the engineering design in various architectures.

Geospatial data with its associated metadata motivated this work, but other types of data now have associated separate metadata files. Currently, if data sets and their metadata are both stored in a data system, there is no way to create a relationship between them and use this knowledge. In addition, data systems are now including ontologies for query expansion, but certain ontologies are only relevant to particular data sets and/or to particular attributes. This more precise knowledge of relationships should be available in a high level manner.

We showed the advantages of the meta ontology by posing queries that allowed the user to not know as much as would otherwise be needed to pose queries. For example, the names of files and attributes are not needed because they can be inferred. This capability would be useful in current DBMSs, even without additional types of data components. As verification, we used SPARQL and Twinkle to run the queries.

We showed that searching for geospatial data can be done by automatically finding the metadata files to search, which also eliminates the need to put selected metadata fields into one large database, as is currently done in geospatial clearinghouse portals. The ability of the user not to have to say which ontology to use for query expansion, as is currently needed in Oracle, for example, was also shown. Finally, we showed how it is possible to expand search (or query) over the content in geospatial data sets themselves, rather than just being limited to searching metadata files to find data files. This enables values in actual data sources to be included as search criteria, which will likely offer more precision in searching. The latter example also showed how to do keyword search in a database, without knowing the schema.

Future work includes extending the model for a large cyberinfrastructure that would include many more kinds of data as well as processes.

We believe that geospatial information systems, DBMSs, cyberinfrastructures, portals, and other types of architectures now need to accommodate various types of data components at a high level of abstraction. Similar to the evolution of a logical data model for Relational DBMSs to hide physical storage details from the user, a data component ontology for information systems provides another needed logical abstraction as more types of data become part of a data system. This work helps provide management and query of geospatial data and associated files.

Acknowledgements. This work was partially supported by NSF's Science and Engineering Information Integration and Informatics (SEIII) program and NSF's Office of Cyberinfrastructure INTEROP program.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American*, 29–37 (May 2001)
2. Das, S., Chong, E., Eadon, G., Srinivasan, J.: Supporting Ontology-based Semantic Matching in RDBMS. In: Nascimento, M., Ozsu, M., Kossmann, D., Miller, R., Blakeley, J., Schiefer, K. (eds.) *Thirtieth International Conference on Very Large Data Bases*, Toronto, Canada, pp. 1054–1065. Morgan Kaufmann (2004)
3. Data.gov, <http://www.data.gov/semantic/index>
4. Dublin Core Metadata, <http://dublincore.org/metadata-basics/>
5. Egenhofer, M.: Toward the Semantic Geospatial Web. In: Voisard, A., Chen, S.C. (eds.) *ACM-GIS 2002*, McLean, VA, pp. 1–4 (November 2002)
6. FGDC Metadata, <http://www.fgdc.gov/metadata/geospatial-metadata-standards>
7. FGDC Metadata Working Group meeting (June 22, 2010), Robert Dollison data.gov presentation
8. Fonseca, F., Davis, C., Camara, G.: Bridging Ontologies and Conceptual Schemas in Geographic Information Integration. *GeoInformatica* 7(4), 355–378 (2003)
9. Geospatial One-Stop, <http://www.geodata.gov>
10. Goodwin, J., Dolbear, C., Hart, G.: Geographical Linked Data: The Administrative Geography of Great Britain on the Semantic Web. *Transactions in GIS* 12(s1), 19–30 (2008)
11. Halevy, A., Franklin, M., Maier, D.: Principles of Dataspace Systems. In: Vansummeren, S. (ed.) *Twenty-Fifth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Chicago, Illinois, June 26–28, pp. 1–9 (2006)
12. Hochmair, H.H.: Ontology Matching for Spatial Data Retrieval from Internet Portals. In: Rodríguez, M.A., Cruz, I., Levashkin, S., Egenhofer, M. (eds.) *GeoS 2005*. LNCS, vol. 3799, pp. 166–182. Springer, Heidelberg (2005)
13. Hydrology Ontology, Ordnance Survey, <http://www.ordnancesurvey.co.uk/oswebsite/ontology/Hydrology/v2.0/Hydrology.owl>
14. Linked Open Data, <http://linkeddata.org/>

15. Ma, L., Wang, C., Lu, J., Cao, F., Pan, Y., Yu, Y.: Effective and Efficient Semantic Web Data Management over DB2. In: 2008 ACM SIGMOD International Conference on Management of Data, New York, NY, pp. 1183–1194 (2008)
16. Olken, F.: An Ontology of Measurement Units and Dimensions (2009), http://ontology.cim3.net/file/work/OntologySummit2009/OntologySummit2009_Symposium_20090406-07/units-ontology-talk-v01-FrankOlken_20090406.pdf
17. Oracle Semantic Technologies (2008), <http://www.oracle.com/technetwork/database/options/semantic-tech/index.html>
18. Protégé, <http://protege.stanford.edu/>
19. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. W3C Recommendation (January 15, 2008), <http://www.w3.org/TR/rdf-sparql-query/>
20. Rodriguez-Muro, M., Lubyte, L., Calvanese, D.: Realizing Ontology Based Data Access: A Plug-in for Protégé, <http://www.inf.unibz.it/~rodriguez/papers/rodr-luby-calv-IIMAS-2008.pdf>
21. Schuurman, N., Leszczynski, A.: Ontology-Based Metadata. *Transactions in GIS* 10(5), 709–726 (2006)
22. Seltzer, M.: Beyond Relational Databases. *ACM Queue*, 50–58 (April 2005)
23. Smith, M.K., Welty, C., McGuinness, D.L. (eds.): OWL Web Ontology Language Guide, W3C Recommendation (February 10, 2004), <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>
24. Tang, W., Selwood, J.: *Spatial Portals, Gateways to Geographic Information*. ESRI Press, Redlands (2005)
25. Theobald, A., Weikum, G.: The XXL Search Engine: Ranked Retrieval of XML Data using Indexes and Ontologies. In: *ACM SIGMOD 2002*, Madison, WI, June 4-6, p. 615 (2002)
26. Twinkle, <http://www.ldodds.com/projects/twinkle/>
27. Uschold, M., Gruninger, M.: Ontologies and Semantics for Seamless Connectivity. *SIGMOD Record* 33(4), 58–63 (2004)
28. Weikum, G., Theobald, M.: From Information to Knowledge: Harvesting Entities and Relationships From Web Sources. In: *PODS 2010*, June 6-11, Indianapolis, Indiana (2010)
29. Wiegand, N.: Semantic Web for Geospatial E-Government Portals. In: Cushing, J., Pardo, T. (eds.) *8th Annual International Digital Government Research Conference (dg.o 2007)*, Philadelphia, PA, pp. 298–299 (May 2007)
30. Wiegand, N., Garcia, C.: A Task-Based Ontology Approach to Automate Geospatial Data Retrieval. *Transactions in GIS Special Issue on the Geospatial Semantic Web* 11(3), 355–376 (2007)
31. Wiegand, N.: Exploring RDBMS Support for Ontology Enhanced Searching in Geospatial Portals. In: *GIScience 2008*, Park City, Utah, pp. 196–200 (2008)

Preserving Detail in a Combined Land Use Ontology

Nancy Wiegand

University of Wisconsin-Madison, Madison, Wisconsin
wiegand@cs.wisc.edu

Abstract. Resolving land use codes between jurisdictions has been an on-going problem due to differences in terms and the nuances of partial similarity of concepts. This paper reports on creating a land use ontology that, contrary to being limited to the highest level of codes or to the most-often used codes, retains all codes. It is also novel in that it records the more subtle relationships between codes rather than just using subclassing. The purpose of creating this comprehensive type of ontology is to provide precise answers to searches of heterogeneous land use codes across jurisdictions. Land use affects important planning decisions, and detail is critical. To query the ontology, custom Java code was written, rather than using SPARQL, to be able to traverse down or up the tree to find the closest matching code when an exact match does not occur.

Keywords: Land use codes, ontology, Semantic Web, programming.

1 Introduction

Land use is an example of a geospatial domain in which there do not tend to be state-wide or national standards. Instead, individual jurisdictions, such as cities, counties, and regional planning commissions, create their own land use coding systems to describe their particular local or regional land uses. Land use codes may be recorded as an attribute as part of individual parcel descriptions, that is, occur in parcel databases, or sometimes land use is recorded in separate shapefiles as polygons forming contiguous land uses, similar to the representation of land cover data. Land use coding systems are typically hierarchical with top levels such as agriculture or residential and subcategories such as cropland or single family. Land use differs from land cover in that land use describes how the land is used rather than describing its vegetative cover, for example. Land use code systems range from extensive detail to a minimal list. Example residential codes are given in the Appendix.

Because of the local jurisdictional differences in land use codes, it is currently difficult to do land use queries over geographic areas that cross jurisdictional boundaries. For example, it is hard to find the cumulative assessed value for each type of land use over multiple counties because the categories of land use vary.

Various approaches have been tried to solve the problem of semantic heterogeneity between land use codes. One solution is to do 'one time' matches between code systems involved in a particular application. This approach requires someone making comparisons by hand for the current application, usually just across a couple code

systems. The resulting matches are not typically reused, nor is there usually a way to make the result mapping known to others for potential reuse.

Another possible solution that is more recently being used in other domains to achieve semantic interoperability is to create a global or reference ontology that covers the conceptual space usually in a high level or general manner or using certain agreed upon categories. With this approach, various heterogeneous terms can then be resolved across applications by individually matching each set of terms to the reference ontology. This is contrary to the above solution in which each code set is matched to another code set rather than to a reference ontology.

Creating a reference ontology works well in some domains especially if domain experts can come to agreement on a selected subset of concepts. The problems with this approach, however, include the generality of the ontology, which likely means losing the detail of the local code sets because many terms are lumped into a few high level categories. Also, there is the difficulty of agreeing on what concepts and terms to include in the ontology.

As a conclusion, solutions to compare heterogeneous code sets in other domains do not meet the needs regarding land use codes. 'One-time' solutions are time-consuming and limited in scope, both in the numbers of codes compared and in geographic extent. Also, reference ontologies have not been created for land use. In any case, a general or global land use ontology would only be useful for comparing broad categories. Further, land use codes are used in making important decisions about land in planning and assessment, for example. Because of this, as much detail as possible should be preserved.

In this paper, we take an ontology approach that is different from the above solutions. We preserve detail by creating a comprehensive combined land use ontology by merging all the local codes into the final ontology. Although this effort is initially time-consuming and difficult, it is established once and is then available for a myriad of future uses. Keeping all codes eliminates the need for domain experts to meet and agree on terms for a land use ontology, which is often difficult.

As part of merging, we started to incorporate various types of relationships, not just sub or super classing, to add more knowledge as to how the codes are related to each other. In this way, our ontology is becoming an ontology in the true sense of being an interconnected graph that forms a complete knowledge base that holds all codes and their inter-relationships. Such background knowledge bases will help form a geographic linked data cloud for land use that could stand alone or be linked to the Linked Open Data Cloud [7]. Such knowledge will contribute to building the Semantic Web. A small example of the type of ontology we are creating is shown in Fig. 1. Additionally, codes may be connected between subtrees.

To search for a land use code in our ontology that combines jurisdictions, it was necessary to write custom Java code rather than using the SPARQL RDF query language [19]. This is because we want our output to return the closest term to the query term, along with its relationship, rather than not returning any results if an exact match does not occur. To do this involves programming to traverse down or up the ontology until certain conditions are met.

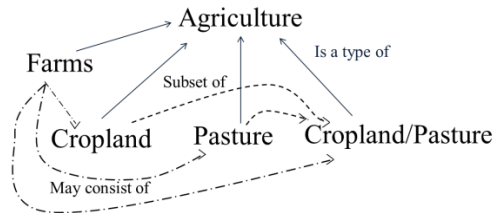


Fig. 1. A small ontology with relationships additional to subclassing

A purpose of this work, in addition to creating a land use ontology, is to present the idea of creating a combined ontology to accomplish semantic interoperability rather than creating a reference ontology or standard that is only a subset of the space. In addition to the combined hierarchy, we started adding meaningful relationships to create a knowledge base. We anticipate that this idea could be applied to other geographic domains when appropriate such as when it is difficult to create a standard or achieve agreement and when it is necessary to preserve local detail.

In the rest of this paper, Section 2 gives related work. Decisions involved in creating the combined ontology are discussed in Section 3. Section 4 presents our software design to access the ontology and shows sample output. Section 5 gives a discussion, and Section 6 has a summary.

2 Related Work

There is no standard land use coding classification. As a result, land use codes are very heterogeneous between jurisdictions. The American Planning Association developed the Land Based Classification Standards (LBCS) to broaden land use to cover various dimensions, such as activity and function [3]. However, LBCS has not been broadly adopted.

Not much work has been done on semantics or ontologies for land use codes. Work on land cover/land use is the closest related work although land cover categories are much more broad because usually a country level area is being categorized, as compared to parcel level local codes. But, similar to land use, land cover classifications are mainly hierarchical and also vary across agencies and countries defining them and through time. For example, in the U.S., the USGS has a 1992 as well as a 2001 National Land Cover Data (NLCD) set. Others have worked on comparing land cover classifications. Work has been done on automated similarity measures [1, 2]. In Gahegan et al., web services are used to mediate between legends of different land cover data sets [8]. The approach taken in this paper is to create a large land use ontology that keeps detail along with nuanced associations. This ontology can then be used as a method for semantic integration to resolve code differences. It is not a standard but serves as a reference knowledge base.

Prior work of the author involved designing and building a query module for resolving heterogeneous land use codes across jurisdictions in Wisconsin. This was a complete end-to-end system [22] embedded into the Niagara XML DBMS [14]. Users

selected a land use code, and, through query re-writing that consulted mappings, subqueries were generated for local parcel data and sent into the Niagara query engine. With access to the local parcel data, aggregated values such as number of acres having a particular land use could be returned. In addition, the locations of actual parcel polygons having a specified land use code were shown on a map. However, in that project, query terms were limited to a drop down list, and query expansion could only retrieve subclasses. This new work, although not an end-to-end system, builds on the prior work to allow a user to query any code at any level of the hierarchy. Resolution of codes across jurisdictions is now done using a full land use ontology that includes synonyms and nuanced relationships, not just subclassing.

There are a number of large projects in the geospatial domain using ontologies to resolve semantics, e.g., CUAHSI [6], iPlant [11], ICAN [10], and Mercury [16]. Some of these ontologies, however, relied on experts meeting to agree on select terms. Instead of that approach, we are keeping all local terms but adding relationships.

Regarding using automated alignment or merging software [e.g., 4, 5, 17] to create the combined ontology, we found that string matching is not reliable regarding the meaning of land use codes. For example, even a more straightforward code, such as ‘Single Family’, could involve mapping choices that include ‘Single Family + Garage’, ‘Single Unit in a Duplex’, and ‘One Family Unit’. Here, for example, the string ‘Single’ does not always occur in related codes or may mean something different. In prior work, we ran several land use code sets through the 2007 version of the AgreementMaker [21]. Although the results were quite interesting and worthwhile, much hand labor was still needed to match code sets. Thus, to create our initial combined ontology, we matched by hand, but, in the future, to add additional code sets, we will likely use merging and/or alignment software to help with a first pass, now that we have a strong base ontology to which most codes will match.

3 Combining Land Use Codes to Make a Comprehensive Ontology

In this section, we discuss the effort and decisions made to create a combined land use ontology that incorporated all the codes. All detail was kept rather than only using a subset of select codes as may be done in other domains. Also, we started to explicitly record nuance relationships in addition to subclass relationships. A large part of the work was human decision-making as to how codes related to each other; this cannot be done automatically when first building the combined ontology.

3.1 Overview

For an example geographic area, we used the state of Wisconsin and obtained code sets from cities, counties, or regional planning commissions (RPCs) (Fig. 2). Each RPC or other jurisdiction developed its own land use codes to classify how land is used in that region. The different ways in which land has been classified across Wisconsin make cross referencing between these regions very difficult.

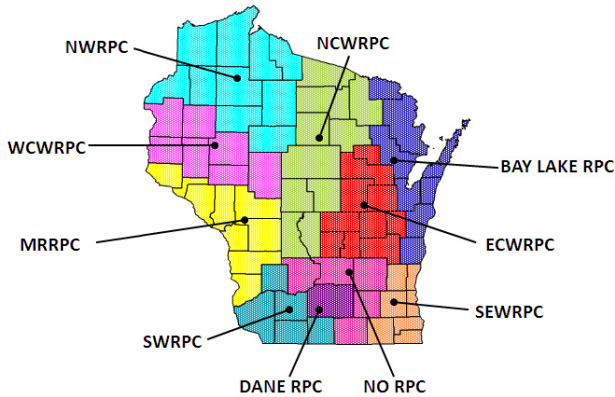


Fig. 2. Nine regional planning commissions in Wisconsin (excluding five counties)

The object in creating our ontology was to combine all the land use codes into one large set much like a tree, with parent and children nodes. Land use codes are mostly hierarchical, but we allowed for multiple parents and other kinds of relationships. Initially, we experimented with Protégé (<http://protege.stanford.edu/>) and associated merging software, but due to string matching issues along with our need to learn the semantics of the codes ourselves and for simplicity, we worked in Microsoft Word, using indented bullets to show subclassing. The resulting Word file is 34 pages long, single spaced. We merged over a thousand codes into a framework of subclasses. Each of the nine main classes has up to five levels of subclassing. The simple Word format made it easier for the combined ontology to be read into a Java program for further processing. Otherwise, OWL [15] code would have needed to be parsed. Now, with the main part of the ontology established, we are recording the ontology in Protégé and will put it into the SOCoP Open Ontology Repository for geospatial ontologies (<http://socop.oor.net>) making it available for others to use. A partial screenshot is shown in Figure 3.

3.2 The Process of Creating a Combined Ontology and Decisions Made

To create a combined land use code ontology, many decisions had to be made about how the classes would be organized. At the beginning, four of the eventual seven land use code sets were available to us (Table 1). These were code sets for the Bay Lakes Regional Planning Commission, the city of Madison, the county of Dane, and East Central Wisconsin Regional Planning Commission. Looking at all four, we decided on nine main classes to become the top-most parent nodes of the ontology. These are: **Residential, Commercial, Industrial, Transportation, Communication/Utilities, Institutional/Governmental Facilities, Recreation, Agriculture/Silviculture, and Natural Areas.**

Table 1. Land use code sets

Code set	Acronym or short name
Bay Lakes Regional Planning Commission	Bay Lakes
City of Madison	Madison
County of Dane	Dane
East Central Wis. Regional Planning Commission	ECWRPC
North Central Wis. Regional Planning Commission	NCWRPC
South East Wisconsin Regional Planning Commission	SEWRPC
County of Eau Claire	Eau Claire

Each of the first four code sets is very different from each other, but we wanted to choose one to serve as a basic backbone of the ontology. We decided to use Bay Lakes as the basic structure because it had the most obvious and easy to read structure out of the initial four sets. It was easy to see the subclasses and superclasses without much familiarity with the code set itself. As such, the entire Bay Lakes code set was the first of all the code sets to be incorporated into the ontology.

The next step was to match the other code sets to the Bay Lakes code set. The Madison and Dane County code sets, however, were much more detailed than Bay Lakes. Incorporating the Madison codes into the ontology was a long process. Madison has over 1000 separate codes due to continued refinement of the codes from a single digit up to 4 digits (e.g., 1 digit [1 Residential]; 2 digit [11 Household units, 12 Group quarters, 19 Other residential, NEC]; 3 digit [e.g., 111 One-family unit, 112 Two-family unit, etc.]; 4 digit [e.g., 1511 Hotels and motels, 1512 Bed and Breakfast, etc.]).

To incorporate Madison and Dane County codes into the ontology, many new, ever more detailed, classes were created. Some of the Madison codes matched the Bay Lakes classes, but many had to become their own classes, being super and sub-listed appropriately. A couple issues included repeats of codes as specificity increased, such as the Madison class name ‘Other Resource Production and Extraction’ which appears three times in the code set under three separate codes: 89, 890, and 8900. Another example was the over-use of ‘Other’. It was common to find a set of subclasses in the Madison code set whose last class was ‘Other’. Further, in some cases, this ‘Other’ class has its own set of subclasses that might include an ‘Other’ category, essentially creating an ‘Other’ subclass of an ‘Other’ class. It was decided to absorb this second ‘Other’ into the first ‘Other’ class to stop this pattern.

The third code set to be merged was Dane County. Because of the similarity between Madison and Dane County codes, it was much easier to see where to fit in Dane County codes. A difference between the Madison and Dane County code sets, however, was that Dane County used the same code for many items (i.e., for a set of items). For example, Dane has one code 53 (‘General Merchandise’) that consists of ‘Department Stores’, ‘Mail Order Houses’, ‘Limited Price Variety Stores’, ‘Merchandise Vending Machine Operators’, ‘Direct Selling Organizations’, and ‘Other Retail Trade - General

Merchandise, N.E.C.’ Contrary to this, Madison has code 53 (‘Retail trade – general merchandise’) but also 531 for ‘Department stores – retail’, 532 for ‘Mail order houses – retail’, etc. To be able to find, for example, ‘Department Stores’ in Dane County in the combined ontology, (Dane, 53) was listed under ‘Department Stores’. However, the ‘53’ needs to be flagged as a set-valued code because it contains many other items than just ‘Department Stores’.

ECWRPC was the fourth code set incorporated. Much like Dane County, the ECWRPC grouped many items into one code, but with ECWRPC, the codes did not fit into the same superclass in the combined ontology. For example, the ECWRPC set-valued code 9441 is a grouping of ‘Apartments’, ‘Three or More Households’, ‘Condos’, and ‘Rooming and Boarding Houses’. These terms were already in the combined ontology but with different superclasses. ‘Apartments and Condos’ is already a category, but ‘Three or More Households’ is under ‘Multiple Family/Three or More Family’. And, there was already a ‘Rooming and Boarding Houses’ class under Commercial (versus Residential). So, the individual codes ended up split under different minor and major superclasses.

The fifth data set acquired was NCWRPC, which was very general and extremely short, consisting of just twelve codes. Most of these were of the nine upper-most classes mentioned earlier. No new classes had to be created to incorporate this region into the combined ontology.

The next code set incorporated into the combined ontology was the SEWRPC codes. This code set was also not extremely difficult to incorporate because by now there were many classes to which the SEWRPC codes matched. However, there were a few new subclasses that needed to be added, such as Public and Private and Local and Regional. Also problematic is that SEWRPC combines service and sales into ‘Retail Service and Sales’ under Commercial, whereas other code sets separate service from sales. Both ‘Retail Sales’ and ‘Retail Services’ already had their own long list of subclasses in the combined ontology, so they could not be combined in anyway. Further, SEWRPC distinguishes Intensive (210) and Nonintensive (220) subclasses for ‘Retail Service and Sales’. To handle this, we decided to make separate classes for Intensive and Nonintensive under each of the existing codes of ‘Retail Sales’ and ‘Retail Services’. This meant that SEWRPC’s service and sales codes were duplicated. If we had been able to obtain written definitions for these two terms, we may have done some orderings differently, but this was the best we could do.

The last code set incorporated into the ontology was the Eau Claire region. This set presented few new issues, as many of the codes simply needed to be added to their appropriate classes in the combined ontology. There were a few instances in this code set, however, where the Eau Claire region did some grouping that needed to be split in order to fit properly into the combined ontology. There was also the situation with mobile home parks. Eau Claire gave each mobile home park in the region its own separate code. Although an ontology such as we are creating should not really contain particular instances of this sort, we did create separate classes for these mobile home parks to satisfy local queries, as per the local land use code set.



Fig. 3. An example of some merged Residential codes in Protégé

3.3 Discussion for Creating a Combined Ontology

Our initial effort was a large and difficult task and could not have been done automatically. Also, in principle, an ontology, if complete, should end up with the same final organization regardless of which code set was used to start building it. But, it is possible that a different order of entry of code sets may have influenced some subtrees. In the end, the ontology will be more of a graph, and the hierarchy aspect will be less important.

The initial approach combined all the codes by fitting each code as a subclass somewhere in the existing structure without considering further relationships, other than synonyms. Many codes fit nicely into a hierarchy, and for some applications, subclassing may be adequate. And without software to retrieve and analyze other kinds of relationships, subclassing is the most practical.

As we worked further in creating the comprehensive ontology, we found that we needed to model multiple parents, set-valued codes, and relationships other than subclassing. Modeling multiple parents is needed especially depending on the use of the ontology. For example, if the use is primarily commercial, many areas under Recreational/Entertainment could also be linked as a subcategory of Commercial because they have a commercial aspect. An example is golf courses which are listed under Recreational. However, they have a role in the commercial sector by providing a service for golfers to use their land, and they sell golfing related items, as well as food and beverages for their customers.

The need to mark some codes as set-valued was discussed earlier. Our solution was to repeat a set-valued code wherever it was needed. We chose to return a superset of information and indicate that to the user rather than miss some information.

As to additional relationship types, in our second approach, we took a more detailed look at each code and its relationship to other codes. Fig. 1 gave an example of the result of this approach. Our software allows for any named relationship. As to

deciding what relationships to model, we could limit relationships to those of SKOS (<http://www.w3.org/TR/2009/REC-skos-reference-20090818/>). Or, we could develop a myriad of types of relationships, such as ‘possiblyTheSame’, ‘likelyTheSame’, ‘relatedSubsetOf’, ‘duplicatedSubsetOfBoth’, ‘hasSomeRelationTo’ (as a bit weaker than ‘relatedTo’) and other kinds of connecting relationships. We found that a compromise approach was the most practical. Just using simple relationships was too limited, but if we incorporated too many precise kinds of relationships, the graph became too difficult to understand and work with. However, we do think relationships that connect codes between subtrees are worthwhile, such as a ‘work’ relationship connecting ‘Farm Residence’ under Residential with ‘Home Occupation in Farm Residence’ under Commercial. Relationships such as these help turn the tree into a graph (in addition to modeling multiple parents). We also propose that additional subtrees could be formed to group codes, such as, under Residential to split codes into those ‘Related to Farms’ versus those ‘Not Related to Farms’. In this way, multiple kinds of subtrees could exist over the same basic set of codes. We did some of these kinds of relationships but anticipate that more will be set incrementally as people use the ontology for various applications.

Our intent here is to create a prototype so that domain people get an idea of the potential of using an ontology to resolve land use codes. Rather than waiting for domain experts to assemble and agree, which is difficult for busy professionals, we created an ontology with enough detail to avoid contention on content. In choosing subtrees and relationship types, we made educated decisions, which could be changed, and we intend to allow domain experts to vet the decisions made. We anticipate that adjustments will be done over time, given the size and complexity of the ontology.

4 Querying the Ontology

The motivation for this work is to solve the problem of resolving heterogeneous land use codes across multiple jurisdictions. We wrote custom Java code to access and traverse the ontology to find the closest matching code.

4.1 Scenario

A typical query is “Find all the parcels in multiple counties that have a particular type of land use, such as cropland”. Our system, written in Java, inputs a land use code and the jurisdictions over which to query. Our program returns each jurisdiction along with its closest matching code to the query term and the relationship of that code to the query term. The result may or may not be an exact match. The local value (e.g., 111) for the closest match is also returned (to be able to later search for parcels with that code value in the appropriate parcel data set).

An empty result will not occur for any input area because, if there is not an exact match, we find the closest matching code. For example, the code ‘cropland’ may not exist by itself in some particular county but may be combined with another code, such

as ‘cropland/pasture’. In this case, cropland/pasture would be returned for that county and noted as being a superset of the query term. This level of detail enables decision-makers to have precise information.

4.2 The Software

Although not implemented as such, the conceptual design for the main data structure in the software is mostly a tree. This is because many of the land use coding systems are organized hierarchically. But, the storage structure is not limited to a strict tree; we allow for multiple parents. We also programmed to accommodate many different kinds of relationships other than parent/child (superclass/subclass) relationships. Other kinds of relationships are not limited and may be ‘synonym’, ‘similar to’, ‘superset’, ‘associated with’, and others. We allow any possible relationship name.

As currently implemented, each main class object in the data structure holds a string for a land use code along with a linked list of all areas (jurisdiction names) containing that code. The actual alphanumeric value of the code for each code set is also stored with each area. For example, the main class node for the code Group Quarters contains the string ‘Group Quarters’ along with a linked list of jurisdictions with their actual code values, such as (Bay Lakes, 170), (ECWRPC, 942), etc. Also, in keeping with the basic tree structure, each class object has pointers to a parent node and to an arbitrary number of children. Therefore, subclass and superclass relationships are inherent in the tree structure. Multiple parent nodes are handled separately. The main class node also stores synonyms for that code. In addition, an arbitrary number of other kinds of relationships (other than subclass and superclass) are stored in each object.

The program fulfills the project’s goal to create and access a knowledgebase where a user can search for land use codes and get results based on relationships with other categories. The search algorithm is explained using an example. Suppose the query is to search for the category ‘Apartments and Condos’ in the region ‘Madison’. After finding the node with the code ‘Apartments and Condos’, a check is done to see if the query region is listed for that code. If it is not found, a search is done for the region in the synonym list. If not found there, the list of other kinds of relationships is checked next. If the region still is not found, the tree is searched downward for subclass terms that may contain the region ‘Madison’. Downward (and upward) search is done recursively using the inherent parent/child relationships. The downward search continues lower and lower until it either finds ‘Madison’ or hits the bottom of the tree. If the region is still not found through the downward search, an upward search is started, again recursively going up until a main category is found, e.g., Residential.

4.3 Query Output

This section shows the input and output for queries. Input consists of a land use code and list of jurisdictions. Output returns the closest code to match the query term along with the relationship and the local numerical or alphabetical value in each jurisdiction’s code set.

1. The following query to find ‘Group Quarters’ over all the data sets shows results for exact matches, superclass, and subclass relationships. As shown, for Eau Claire and NCWRPC, the main category of Residential is returned because the specific code ‘Group Quarters’ does not occur. Contrary to that, for SEWRPC, ‘Group Quarters’ has subclasses, which are returned.

```
Input category: Group Quarters
Input each area. Type "done" when done
All
done

Bay Lakes (170): Group Quarters, exact match
Dane County (129): Group Quarters, exact match
Eau Claire (R): Residential, superclass of Group Quarters
ECWRPC (942): Group Quarters, exact match
NCWRPC (5): Residential, superclass of Group Quarters
Madison (12): Group Quarters, exact match
SEWRPC (661): Local, subclass of Group Quarters
SEWRPC (662): Regional, subclass of Group Quarters
```

2. This next query shows a synonym relationship returned for ECWRPC.

```
Input category: Dorms
Input each area. Type "done" when done
All
done

Bay Lakes (170): Group Quarters, superclass of Dorms
Dane County (129): Group Quarters, superclass of Dorms
Eau Claire (R): Residential, superclass of Group Quarters
ECWRPC (942): Residence Halls, synonym of Dorms
Madison (123): Dorms, exact match
NCWRPC (5): Residential, superclass of Group Quarters
```

3. This last query shows that any other kind of relationship will be returned, here ‘similar to’.

```
Input category: Multiple Family
Input each area. Type "done" when done
Madison
Bay Lakes
NCWRPC
Eau Claire
Dane County
SEWRPC
done

Madison (113): Three or more Family, similar to Multiple Family
Madison (1131): Three Family Unit, similar to Multiple Family
Madison (1132): Four Family Unit, similar to Multiple Family
Bay Lakes (150): Multiple Family, exact match
NCWRPC (6): Multiple Family, exact match
Eau Claire (RM): Multiple Family, exact match
Dane County (115): Multiple Family, exact match
SEWRPC (141): Multi-Family Low Rise: 1-3 Stories, subclass of Multiple Family
SEWRPC (142): Multi-Family High Rise: 4 + Stories, subclass of Multiple Family
```

4.4 Discussion

Our system is designed to be an intercepting subsystem for a database management or other storage and query system that holds the actual parcel data. Our system provides the information to generate subqueries that would then be sent into a query engine to be executed, although we do not yet generate those subqueries. For example, as in Query 3, to find Multiple Family parcels in the NCWRPC and Eau Claire areas, we return the codes ‘6’ and ‘RM’ respectively, which are necessary to query the parcel

data sets. A DBMS with access to the actual parcel data would then be able to return, for example, the total number of acres or total assessed value of lands having a certain land use code.

Custom Java code was needed because, although the combined ontology could be recorded in OWL, an OWL/RDF query language (i.e., SPARQL) is not able to traverse the tree to be able to return the closest matching code when an exact match does not occur. A programming language with conditional statements is needed. This work was inspired, however, by the new modeling capabilities of Semantic Web technologies. Working with OWL ontologies formed the motivation to create an interconnected land use ontology.

As to future work, we are finishing putting the codes and relationships into OWL. That way the ontology will be formally represented and available to all for various uses. We also plan to put the OWL code into Knoodl (knoodl.com) or Collaborative Protégé, for example, so that it is Web accessible and able to be viewed, vetted, or changed by domain experts. For our Java program, we experimented mostly with the Residential subset of codes, so we will work with more of the codes. We may continue the project to be end-to-end and integrate our code with an open source spatial DBMS to be able to retrieve additional parcel information.

5 Summary

Land use is a very rich and important domain. But, land use coding systems are quite diverse. Combining or comparing codes across jurisdictions is extremely difficult. The emergence of the idea of ontologies to store background knowledge inspired this work, but not in the sense of creating a global agreed-upon land use coding standard for which it would be difficult to get agreement and which would lose local specifics. Instead, we present the idea of keeping all detail and storing innumerable kinds of relationships to create an interconnected graph of knowledge of land use categories.

This paper makes a contribution to the area of semantic interoperability by using ontology merging to create a comprehensive domain ontology that is then used to resolve terms. This avoids relying on domain experts for the difficult task of creating a reference ontology from scratch. Domain experts can much more easily vet an existing ontology. Also contention is avoided by including all local codes. The approach of keeping detail and recording relationships can also be applied to other domains.

The value of our system is that it returns the most precisely related terms across the geographic query space. It does so by searching a novel (for land use) combined ontology that is a combination of all land use code sets containing specific inter-relationships.

Our ontology can continue to be expanded to include codes from other states. Initially, our goal is a state-wide land use ontology, but ultimately, it could be a national land use ontology and used for nation-wide land use analyses and decisions. It could be integrated into a National Land Parcel Data initiative [e.g., 13]. Again, the idea is that, instead of having a national land use standard with a limited set of codes, a combined ontology with relationships can be ubiquitous and able to be continually expanded as new code sets are added.

Acknowledgements. This work was partially supported by the National Science Foundation's Office of Cyberinfrastructure (OCI), INTEROP Grant No. 0955816, and a Research Experience for Undergraduates (REU) supplement award. The undergraduates contributing to this work were Joseph Kahl and Karissa Metko at the University of Wisconsin-Madison.

References

1. Ahlqvist, O.: Land Cover/use Legend Harmonization, IIASA 2011 (2011), http://www.iiasa.ac.at/Research/FOR/lc/presentations/Ahlqvist_t_IIASA.pdf
2. Ahlqvist, O., Shortridge, A.: Spatial and Semantic Dimensions of Landscape Heterogeneity. *Landscape Ecol.* 25, 573–590 (2010), <http://www.springerlink.com/content/41076746pu50n054/fulltext.pdf>
3. American Planning Association, Land Based Classification Standards (LBCS), <http://www.planning.org/lbcs/background>
4. Cruz, I.F., Sunna, W., Ayloo, K.: Concept Level Matching of Geospatial Ontologies. In: GIS Planet Second Conference and Exhibition on Geographic Information, Estoril, Portugal (June 2005)
5. Cruz, I.F., Stroe, C., Caimi, F., Fabiani, A., Pesquita, C., Couto, F., Palmonari: Using AgreementMaker to Align Ontologies for OAEI 2011 (2011), http://ceur-ws.org/Vol-814/oaiei11_paper1.pdf
6. CUAHSI, <http://www.cuahsi.org>
7. Cyganiak, R., Jentzsch, A.: Linking Open Data Cloud, <http://lod-cloud.net/>
8. Gahegan, M., Smart, W., Masoud-Ansari, S., Whitehead, B.: A Semantic Web Map Mediation Service: Interactive Redesign and Sharing of Map Legends. In: Spatial Semantics and Ontology Workshop (SSO 2011) at ACM SIGSPATIAL, Chicago, IL, November 1 (2011)
9. GeoSPARQL, <http://www.opengeospatial.org/standards/requests/80>, <http://geosparql.org>
10. ICAN, International Coastal Atlas Network, Ontology Development, <http://ican.science.oregonstate.edu/node/571>
11. iPlant, <http://www.iplantcollaborative.org/discover/semantic-web/semantic-web-overview>
12. Knoodl, Distributed Information Management System, <http://knoodl.com>
13. National Land Parcel Data, A Vision for the Future. National Academies Press, http://www.nap.edu/openbook.php?record_id=11978
14. Naughton, J., DeWitt, D., Maier, D., et al.: others: The Niagara Internet Query System. *IEEE Data Engineering Bulletin* 24(2), 27–33 (2001)
15. OWL 2 Web Ontology Language, <http://www.w3.org/TR/owl2-overview/>
16. Pouchard, L., Cook, R., Green, J., Palanisamy, G., Noy, N.: Semantic Technologies Improving the Recall and Precision of the Mercury Metadata Search Engine. AGU Fall Meeting (2011), <http://m.core-apps.com/agu2011/abstract/ed111cea5ff27cc0c5786310e9462efc>
17. Prompt (Protégé plugin), <http://protege.stanford.edu/plugins/prompt/prompt.html>
18. SKOS, <http://www.w3.org/2009/08/skos-reference/skos.html>
19. SPARQL Query Language for RDF, <http://www.w3.org/TR/rdf-sparql-query/>

20. Sunna, W., Cruz, I.F.: Using the AgreementMaker to Align Ontologies for the OAWI Campaign 2007. In: Second ISWC International Workshop on Ontology Matching. CEUR-WS (2007)
21. Wiegand, N.: A Geospatial Semantic Web Scenario. Technical Report, University of Wisconsin-Madison (2008)
22. Wiegand, N., Zhou, N.: Ontology-Based Geospatial Web Query System. In: Agouris, P., Croitoru, A. (eds.) Next Generation Geospatial Information: From Digital Image Analysis to Spatio-Temporal Databases. ISPRS Book series, pp. 157–168. Taylor & Francis, Balkema (2005)

Appendix: A Few Code Sets Showing Residential Codes

Bay Lakes Regional Planning Commission

100 Residential
 110 Single Family Residential
 111 Single Family Residential Garage
 130 Two Family
 150 Multi-Family
 170 Group Quarters
 180 Mobile Homes
 190 Land Under Residential development
 195 Residential Open Space/Vacant Lot
 199 Vacant Residence

Dane County RPC

111 Single Family

 113 Two Family
 115 Multiple Family
 129 Group Quarters
 140 Mobile Home
 142 Mobile Home Park
 116 Farm Unit
 190 Seasonal Residence

East Central Wisconsin Regional Planning Commission (ECWRPC)

94 Residential, vacated, other or unknown
 9411 Single Family Structures/Duplexes – includes the mowed land surrounding house and Bed & Breakfast Houses
 9413 Farm Residences, includes mowed yard
 9414 Mobile Homes Not in Parks, includes mowed yard
 9416 Accessory Residential Uses/Buildings (ECWRPC CODE) i.e., garages/sheds, includes mowed land surrounding the unit. If the garage is attached to a single family dwelling and is coded 9411 with the house. (sic)
 942 Resident Halls, Group Quarters, Retirement Homes, Nursing Care Facilities, Religious Quarters, includes parking
 943 Mobile Home Parks
 9441 Apartments, Three or More Households: includes condos, Rooming and Boarding Houses – includes parking and yard

Eau Claire County

RB Boarding house
 RC Condominium complex
 RCD Residential unit in condominium, duplex
 RCU Ind unit in condominium complex
 RD Duplex or other two-family residence
 RDU A single unit in a duplex
 RF Four unit dwelling or fourplex
 Etc.

The Maptree: A Fine-Grained Formal Representation of Space

Michael Worboys

School of Computing and Information Science
University of Maine, Orono ME 04473, USA

<http://www.worboys.org>

Abstract. This paper introduces a new formal structure, called the maptree, that is shown to uniquely specify, up to homeomorphism, the topological structure of embeddings of graphs in orientable, closed surfaces. A simple modification is made to show that the representation also works for planar embeddings. It is shown that the maptrees are capable of providing a rich representation of the topology of 2D spatial objects and their relationships. The maptree representation is then used to characterize some properties of topological change in these embeddings.

Keywords: maptree, topology, topological change, geographic information science theory.

1 Introduction

Although there is a considerable body of research on topological relationships between spatial objects (see, for example, [6,3]), there is much less that takes into account the finer topological details of these objects and their relationships. For example, in the region connection calculus, connection between regions at a point, at several points, at a line, or at several lines, cannot be distinguished by the basic theory. Similarly the theory of topological change developed by the author and colleagues in [5,4] presents a theory of topological change that can distinguish, for example, between a hole emerging from a point in the center of a region and the same region merging with itself to create a hole, but cannot distinguish a merge of two regions at a point, or at a linear boundary.

The purpose of this paper is to describe research that brings together two separate formal descriptions of topological configurations, namely combinatorial maps [2,8] and adjacency trees [1,7] to provide a fine-grained representation of spatial objects and their relationships. The structure presented in this paper is an edge-labelled, node-colored tree, that we name a *maptree*. The spatial objects under consideration are configurations of regions, and it is convenient to consider such configurations as embeddings of graphs in orientable closed surfaces or the plane.

A *graph* is defined in the usual way as a set of vertices and edges between vertices. In this paper we allow *loops*, that is edges that connect a vertex to itself, and also multiple edges between two vertices. Often, such structures are referred

to as *pseudographs*, but we continue to use the term “graph” throughout this paper. A graph is *connected* if any pair of its vertices may be linked by a chain of adjacent edges.

Informally, an *embedding* of a graph in a surface is a drawing of the graph on the surface in such a way that its edges may intersect only at their endpoints. The surfaces of interest here are the orientable closed surfaces in \mathfrak{R}^n , and it is well known (Möbius classification theorem for orientable, closed surfaces, 1863) that such a surface is homeomorphic to a sphere with g handles (g -holed torus), for $g \geq 0$. The non-negative integer g is referred to as the *genus* of the surface. We shall usually only be concerned with the sphere (of genus zero). As well as these closed surfaces we also shall consider the Euclidean plane \mathfrak{R}^2 (not a closed surface). From now on, we shall assume all surfaces under consideration are orientable, and omit that term from their descriptors.

Graph embeddings in closed surfaces have the property that the complement in the surface of an embedding of a connected graph is a collection of regions or *faces*, and each of these faces is a 2-manifold. If, furthermore, each of the faces is homeomorphic to a disc, the embedding is called a *2-cell embedding*. When the graph is embedded in the Euclidean plane, then one of the faces will be of infinite extent, and called the *external face*. Of course, not all graphs can be embedded in the plane.

2 Permutations and Combinatorial Maps

In this section we review the basic concepts around the combinatorial map. Such a map provides a unique (up to homeomorphism) symbolic representation of an embedding of a connected graph. By way of introduction, we review some material on permutations.

Let $A = \{a, b, \dots, k\}$ be a finite collection of elements. We call any bijective function $\phi : A \rightarrow A$ a *permutation* of A . Essentially, we can think of ϕ as rearranging the elements of A . Now, any permutation can always be written as a collection of cycles $(a_1 a_2 \dots a_n)$, where $a_2 = \phi a_1$, $a_3 = \phi a_2$, and so on, and $a_1 = \phi a_n$. So, for example, suppose $A = \{a, b, c, d, e\}$, and $b = \phi a$, $c = \phi b$, $a = \phi c$, $e = \phi d$, and $d = \phi e$. Then ϕ may be written in cycle notation as $\phi = (abc)(de)$.

Suppose now that we have a collection of permutations of A , $\Phi = \phi_1, \dots, \phi_m$. Then Φ is *transitive* if, given any elements $x, y \in A$, we can transform x to y by a sequence of permutations from Φ . That is,

$$x \xrightarrow{\phi_{i_1}} x_1 \xrightarrow{\phi_{i_2}} \dots x_p \xrightarrow{\phi_{i_p}} y$$

We have now sufficient preliminaries to define a combinatorial map.

Definition 1. A combinatorial map $M\langle S, \alpha, \tau \rangle$ consists of:

1. A finite set S of elements, called semi-edges, where the number of semi-edges is even. We can write S as $S = \{a, \bar{a}, b, \bar{b}, \dots, k, \bar{k}\}$
2. A permutation α of S

3. A permutation τ of S , which in cyclic form is $\tau = (a\bar{a})(b\bar{b}) \dots (k\bar{k})$ subject to the constraint that $\{\tau, \alpha\}$ is transitive.

An example of a combinatorial map that we shall use again is given by $\mathbf{M}_1\langle S, \alpha, \tau \rangle$, where $S = \{a, \bar{a}, b, \bar{b}, c, \bar{c}\}$, $\alpha = (\bar{a}\bar{c}\bar{b}\bar{c})(a)(b)$, and $\tau = (a\bar{a})(b\bar{b})(c\bar{c})$. It is easy to check that $\{\tau, \alpha\}$ is transitive.

We are now able to relate combinatorial maps to graph embeddings using the following key result due to Edmonds [2] and Tutte [8].

Theorem 1. (Edmonds, Tutte) *Each combinatorial map provides a topologically unique (up to homeomorphism of the surficial embeddings) representation of a 2-cell graph embedding in a closed surface. Conversely, every 2-cell graph embedding in a closed surface can be uniquely (up to permutation group isomorphism) be represented by a combinatorial map.*

We do not repeat a formal proof of this result, but indicate the construction. Given a combinatorial map $\mathbf{M}\langle S, \alpha, \tau \rangle$, the 2-cell embedding is constructed as follows. Each edge of the embedded graph is represented by a pair of semi-edges, called a *facing pair*, transposed by τ . Each cycle of α defines the ordering of semi-edges around each face of the embedding. Each face is defined as the region on the left while traversing the semi-edges of a cycle of α . We may note that the constituent cycles of α are sufficient to uniquely reconstruct the embedding. The constituent cycles of α are termed the α -cycles of \mathbf{M} .

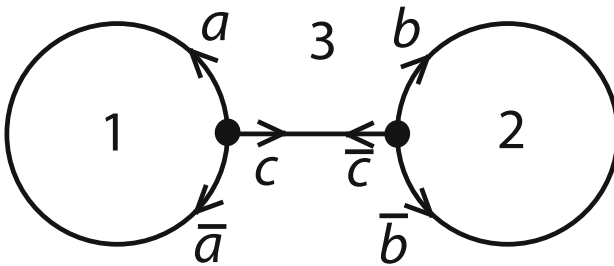


Fig. 1. Embedding of \mathbf{M}_1 in the sphere

Figure 1 illustrates this construction using combinatorial map, $\mathbf{M}_1\langle S, \alpha, \tau \rangle$. We observe that the three α -cycles $(\bar{a}\bar{c}\bar{b}\bar{c})$, (a) , and (b) define the three faces, labelled 1, 2, and 3. For example, the face labeled 1 is defined as the region to the left while traveling the semi-edge a , while the “outer” face labeled 3 is the region to the left while traversing the directed path given by the cycle of semi-edges $\bar{a} \rightarrow c \rightarrow b \rightarrow \bar{c} \rightarrow \bar{a}$.

We note that although the configuration is of necessity reproduced on the plane paper, it is actually a spherical embedding. This can be seen by observing figure 2, where we show two homeomorphic embeddings of $\mathbf{M}_1\langle S, \alpha, \tau \rangle$ in a

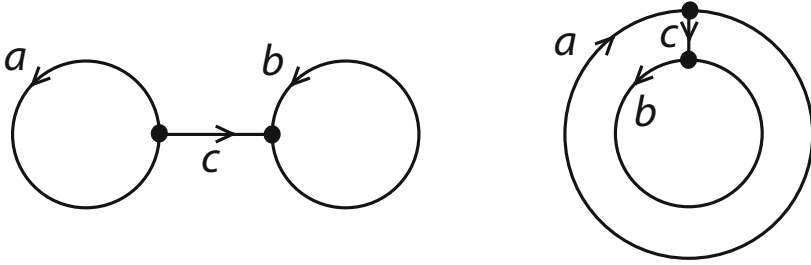


Fig. 2. Two homeomorphic embeddings of M_1 in the sphere

sphere. Note also, for reason of clarity, we show only one of each pair of semi-edges.

We now focus on the nature of the closed surface in which the configuration represented by a combinatorial map is embedded. This surface must be unique by Theorem [1](#). To determine the surface from the combinatorial map, we need to determine the number of vertices of the embedded graph. To do this, we calculate a further permutation β by the formula $\beta = \tau\alpha^{-1}$ where the product is a composition of functions, and α^{-1} denotes the inverse function of α . In our example $\beta = (a\bar{a}c)(b\bar{b}\bar{c})$. Now we note that each cycle of β represents a vertex, and the cycle itself represents the ordering of semi-edges around the vertex.

Now we invoke the famous result of Euler and Poincaré:

Theorem 2. (*Euler-Poincaré*) *Given a 2-cell embedding of a graph in a surface of genus g . Suppose that V , E , and F are the numbers of the embedding's vertices, edges, and faces, respectively. Then:*

$$V - E + F = 2 - 2g$$

Given a combinatorial map, $\mathbf{M}\langle S, \alpha, \tau \rangle$, the genus of its embedding surface can then be calculated as follows.

V = the number of constituent cycles of β

E = the number of constituent cycles of τ

F = the number of constituent cycles of α

and then, by Theorem , the genus g is given by $V - E + F = 2 - 2g$. We sometimes refer to the genus of the combinatorial map, meaning the genus of its embedding surface.

For our example, $V = 2, E = 3, F = 3$, and so $2 - 2g = 2$, and $g = 0$ which accords with our knowledge that the embedding surface is a sphere.

2.1 Planar and Spherical 2-Cell Embeddings

For practical purposes, the embedding surfaces of interest are the sphere (closed surface of genus zero) and the Euclidean plane. The above results on combinatorial maps apply to closed surfaces, and therefore do not apply directly to

the Euclidean plane. However, any graph that is embeddable in the sphere is embeddable in the plane, and conversely. The Euclidean plane is topologically equivalent to a punctured sphere. Suppose we have a map that is 2-cell embeddable on the surface of a sphere. We now puncture the sphere, taking care that the puncture does not lie on an edge or vertex of the embedding. (The cases where the puncture lies on an edge or vertex does not concern us, as we are not concerned with unbounded embeddings in the plane.) We now have a planar embedding of the map, and the face in which the puncture occurs is the infinite external face. Of course, the topological nature of a planar 2-cell embedding is dependent upon in which face of the spherical 2-cell embedding the puncture occurs. The plane places an extra piece of structure on the 2-cell embedding in that we have the notion of the infinite face. So, a 2-cell embedding in the sphere may correspond to many topologically distinct 2-cell embeddings in the plane. We can see this by re-examining figure 2, where the left and right embeddings are the same on the sphere but distinct on the plane.

In order that a combinatorial map can uniquely specify a planar 2-cell embedding, all we need to do is specify which cycle represents the boundary of the external face. We indicate the distinguished external boundary cycle in α by square brackets. In our example, the lefthand and righthand planar 2-cycle embeddings are given by $[\overline{ac}\overline{bc}](a)(b)$ and $\alpha = (\overline{ac}\overline{bc})[a](b)$, respectively. We can then invoke a slightly extended version of Theorem 1 (Edmunds, Tutte) to guarantee topological uniqueness of the representation.

3 Representations of Non-connected Graph Embeddings

Up to now, the focus has been on connected graphs, because only connected graphs have 2-cell embeddings and are representable by combinatorial maps. In this section we extend the algebraic representation of graph embeddings as combinatorial maps to also represent embeddings of non-connected graphs. In our examples, we focus on the Euclidean plane and sphere, as these are the important cases in practice. However, our results carry over to surfaces of any genus.

When we remove the constraint that the graphs are connected, we have to add some extra structure to the algebraic representation. To see this, consider the two graphs, each embedded in the surface of a sphere, shown on the lefthand and righthand sides of figure 3. Both embeddings would have the same representation as a collection of three combinatorial maps. However, it is easy to see that these embeddings are not topologically equivalent, in the sense that there does not exist a homeomorphism of the sphere that maps one embedding to the other. How graph embeddings stand with respect to one another becomes an issue, and we cannot just represent the embedding of a non-connected graph as a set of maps. In this section we develop the extra structure needed to provide topologically unique representations for planar and spherical embeddings of non-connected graphs.

As a preliminary to our main result, we revisit a known representation of a collection of closed curves embedded in a surface as a tree.

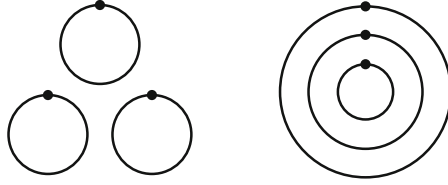


Fig. 3. Two non-homeomorphic embeddings in the sphere

3.1 Regions Created from Nested Closed Curves

Consider a closed surface in which is embedded a disjoint collection of closed curves. An example of such a configuration is shown in the left-hand part of figure 4. Given any such configuration, we may define an associated tree as follows:

Definition 2. *Suppose given a disjoint collection C of closed curves embedded in a closed surface, such that each closed curve viewed on its own results in a two-cell embedding in the surface. Such a collection of curves partitions the surface into a set of regions. We define the adjacency tree of C as a graph $T_C = \langle N, E \rangle$, where N is a set of nodes, each node representing one of the regions and E is a set of edges. Two nodes are joined by an edge if their associated regions share a common boundary.*

Note that this definition begs the question whether or not T_C is a tree. To see this, consider the number of nodes and edges in the graph. The smallest possible such graph has no edges, and a single node, representing the case where the region covers the entire surface. In the general case, if we add another closed curve (edge) to our configuration, because of the two-cell embedding property of the curve, adding the curve adds one more region (node) to T_C . So, by induction, T_C has one more node than edges. Because T_C must be connected, it follows that T_C is a tree.

It is known that that an adjacency tree characterizes the configuration of regions in the plane uniquely, up to homeomorphism from the surface to itself (see, for example, [17]).

Note that if the embedding is planar, then we can distinguish the infinite external region by making the node representing it in the adjacency tree the root of a rooted tree. So, in the example in figure 4, the node labelled 1 will be the root.

3.2 Maptrees

The key insight of research reported in this paper is that a combinatorial map provides a symbolic representation of the 2-cell embedding of a connected graph in a closed surface (a connected, complex structure), while an adjacency map provides a symbolic representation of a collection of closed curves embedded in

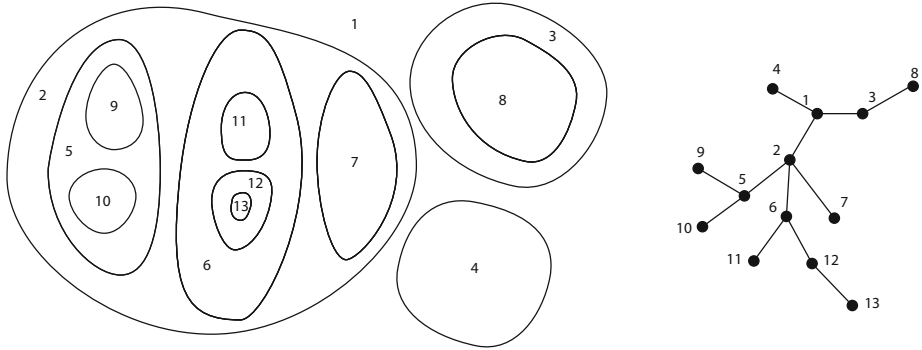


Fig. 4. A collection of nested closed curves and its tree

a closed surface (a non-connected, simple structure). We now show how these constructions can be combined.

Definition 3. A bw-tree X is a colored tree with the nodes colored black or white, respectively, subject to the condition that no two adjacent nodes have the same color. A bw-tree is called a star if it contains exactly one black node.

So as to more easily follow the construction that follows, we illustrate with the embedding shown in figure 5. We assume to begin with that the embedding surface is a sphere. We have omitted the directions of arcs for simplicity.

As a first step, we use a bw-tree to represent the nesting properties of the components of the graph. Figure 6 shows this for our example. The components, labelled M_1, \dots, M_6 in the lefthand side of the figure can each be conceived as a black region, represented by the black nodes in the bw-tree on the righthand

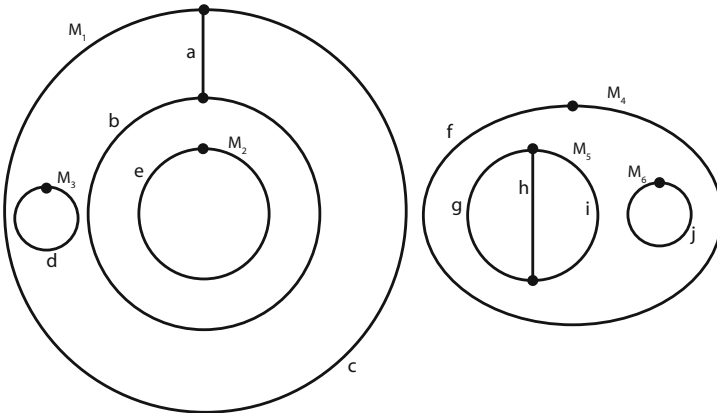


Fig. 5. Embedding of a non-connected graph

side of the figure. The regions labelled $1, \dots, 9$ delineated by the edges of the graph are represented by white nodes in the bw-tree.

This bw-tree uniquely represents how the components and regions stand in the relation to each other, but does not provide details about the topology of the components themselves. To provide this information, we need to consider the combinatorial maps of the components. The graph has six components, and each embedded component M_1, \dots, M_6 considered alone is a 2-cell embedding and so can be represented by a map (using the permutation α to represent each map), as follows:

$$\begin{aligned}
 M_1 &= (a\bar{c}ab)(\bar{b})(c) \\
 M_2 &= (e)(\bar{e}) \\
 M_3 &= (d)(\bar{d}) \\
 M_4 &= (f)(\bar{f}) \\
 M_5 &= (gi)(\bar{g}h)(\bar{h}i) \\
 M_6 &= (j)(\bar{j})
 \end{aligned}$$

The final stage of the construction is shown in figure 7. Each of the components is represented by a star, labelled with α -cycles from its combinatorial map. The stars are joined together in such a way that the edges connecting a white node form the boundary of the region represented by that node. This structure we term a *maptree*.

We are now ready to give the formal definition of a maptree.

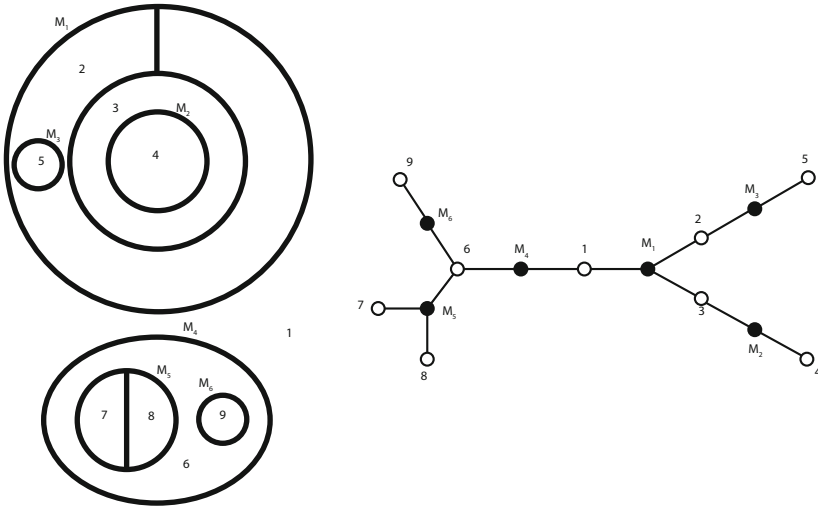


Fig. 6. Region-oriented view of the embedding in figure 5

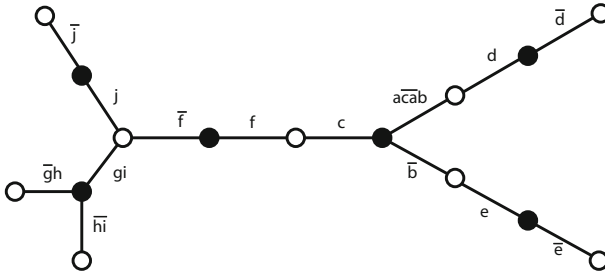


Fig. 7. Maptree of the embedding in figure 5

Definition 4. Let \mathcal{M} be a finite collection of combinatorial maps. A maptree is an edge-labelled bw-tree $\mathcal{T}_{\mathcal{M}}$ such that the edges of each star are labelled by the α -cycles of one of the combinatorial maps in \mathcal{M} .

It is clear from the above example how we should interpret a maptree as a collection of 2-cell embeddings. However, let us just spell out the details. The interpretation of maptree $\mathcal{T}_{\mathcal{M}}$ is that each black node of $\mathcal{T}_{\mathcal{M}}$ represents the 2-cell embedding of the map associated with that node. If two black nodes are connected via a white node, then the two cycles labeling the two edges joining the nodes represent the regions that “face up to each other” in the embedding.

We have the following proposition.

Proposition 1. Let \mathcal{M} be a finite collection of combinatorial maps of genus zero, and $\mathcal{T}_{\mathcal{M}}$ be a maptree. Then $\mathcal{T}_{\mathcal{M}}$ provides a unique representation (up to homeomorphism of the sphere) of the non-connected graph embedding of the connected components represented by \mathcal{M} .

We sketch the proof of this result by indicating the steps in the construction of the graph embedding from a map tree, and note that there are no topological ‘degrees of freedom’ in the process. The process is essentially the reverse of the process described above, where the maptree is constructed from a graph embedding.

Suppose given a maptree $\mathcal{T}_{\mathcal{M}}$, where \mathcal{M} is a finite collection of combinatorial maps.

Step 1: Consider $\mathcal{T}_{\mathcal{M}}$ as specifying the topological configuration of black and white regions on the sphere. As already discussed, such a configuration is unique up to homeomorphism of the sphere.

Step 2: For each black region consider the node n of $\mathcal{T}_{\mathcal{M}}$ that represents it. Then, there is some map $M \in \mathcal{M}$ such that n is said to be the node associated with \mathbf{M} . Replace this region with the graph embedding represented by \mathbf{M} . This replacement also results in a topologically unique embedding.

Step 3: The resulting graph embedding, formed as the composite of all the embeddings in Step 2, is the result, and is unique up to homeomorphism of the sphere.

The proposition can be generalized to the case where the maps in \mathcal{M} embedded into surfaces of any genus. In that case, the genus of the embedding surface of the maptree is the sum of the genera of the constituent combinatorial maps.

3.3 Maptrees for Planar Embeddings

The maptree construction is based on the notions of adjacency trees and combinatorial maps, both of which work for embeddings in closed surfaces. Many practical cases are concerned with embeddings in the plane. In the same way as combinatorial maps are modified, we can modify the map tree construction to account for planar embeddings. As before, we need to distinguish the infinite exterior region, represented by one of the white nodes. We do this by making this the root of a rooted tree. The formal definition follows.

Definition 5. Let \mathcal{M} be a finite collection of combinatorial maps. A planar maptree is an edge-labelled rooted bw-tree $\mathcal{T}_{\mathcal{M}}$ such that the edges of each star are labelled by the α -cycles of one of the combinatorial maps in \mathcal{M} , and the root of the tree is a white node.

As an example, the planar maptree for the planar embedding shown in figure 5 is shown in figure 8, where the root is at the top of the figure.

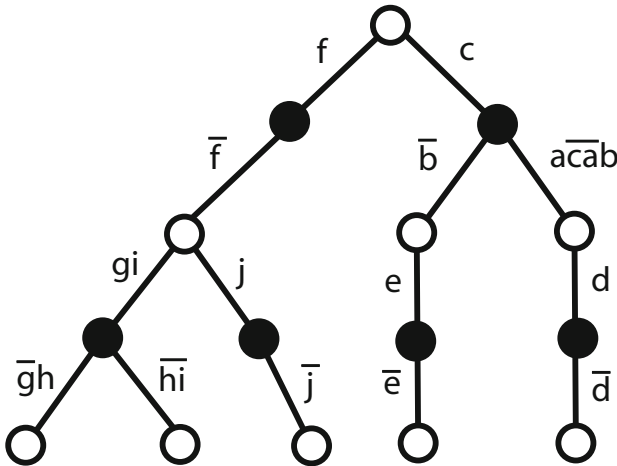


Fig. 8. Maptree of the planar embedding in figure 5

4 Maptrees and Topological Change

The previous sections have shown how the maptree construct provides a topologically complete representation of configurations of regions in a surface. In this section we provide a few examples that demonstrate how maptree operations serve to represent varieties of topological change. We assume that the embedding surface is the plane. However, all the results carry forward to any closed surface. The examples are illustrative of general results, not proved here, but the important concepts are illustrated.

4.1 Merge of Two Regions at a Point

The top portion of figure 9 shows the merger of two regions at a common point, while the bottom of the figure shows the corresponding maptree transformation. We can note that the merger is represented by a folding together of the edges labeled \bar{a} and \bar{b} . This is a general principle that holds for more complex examples – a point merger is represented by a folding about a white node, thus merging two black nodes. The labels of the components of the fold are concatenated. In general, where the labeling is more than a single edge, there are many ways of performing the concatenation of cycles, each corresponding to a distinct point merge.

4.2 Merge of Two Regions at an Edge

The first transition of the top portion of figure 10 shows the merger of two regions using the region edges a and c . Edges a and c merge together forming new edge e . Strictly speaking, we should say that semi-edges a and \bar{c} merge to semi-edge e , and semi-edges \bar{a} and c merge to semi-edge \bar{e} .

The first transition of the bottom part of the figure shows the action on the corresponding maptrees. As with point merge, edge merges are represented

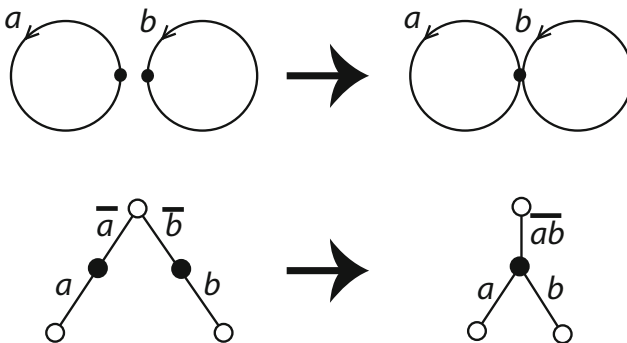


Fig. 9. Two regions merging at a point

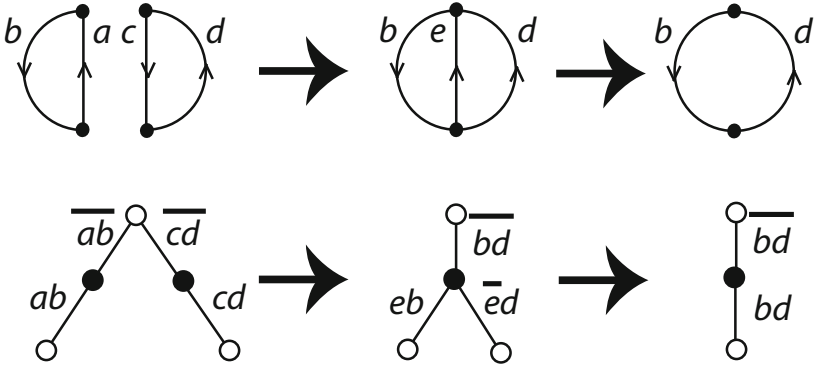


Fig. 10. Two regions merging at an edge and then the edge is deleted

by foldings and concatenations in the maptree. The merge between edges a and c is represented in the maptree as a fold about the maptree root with concatenation of \overline{ab} with \overline{cd} . Now, some algebraic structure comes into play. The elements a, b, c, d, e, \dots , and $\overline{a}, \overline{b}, \overline{c}, \overline{d}, \overline{e}, \dots$ generate a group, where the operation is concatenation, the identity is the empty word Λ , and the inverse of x is \overline{x} . In our case, bearing in mind the cycle structure, we have:

$$\overline{b}\overline{a}\overline{c}\overline{d} = \overline{b}\Lambda\overline{c} = \overline{b}\overline{c}$$

because $\overline{a} = c$ and so $\overline{a}\overline{c} = \Lambda$. Also, we substitute e for a in label eb and \overline{e} for c in label cd .

Because the labels on maptree edges are cycles rather than plain words, there are other allowable concatenations, and these will result in other possible transformations of the regions.

4.3 Edge Deletion

Both types of merge in the preceding subsection were represented by folding two edges of the map tree together, where the fold was made at a white node. We now give an example of an edge deletion and its representation, which turns out to be a fold about a black node. Continuing on from the merger shown in figure 10, we now delete edge e , as shown in the second transition of the top portion of figure. The second transition of the bottom part of the figure shows the action on the corresponding maptrees. In this case, the edges labeled eb and $\overline{e}d$ fold into the single edge labeled bd . This label also follows the operations of edge group, as defined in the previous section, where:

$$b\overline{e}d = b\Lambda d = bd$$

5 Conclusions

This paper has developed a topological representation of closed surficial embeddings. We have demonstrated that the representation does indeed contain sufficient information to generate unique embeddings, up to homeomorphism of the embedding surface. We have also shown how a simple modification provides a representation for planar embeddings.

One strand of this research continues earlier work on spatial relationships (see, for example, [6,3]). While RCC and the n -intersection methods are able to represent some level of topological detail of the relationship between two regions, the maptree goes further in dealing with multiple regions and providing a full topological representation.

The theory of topological change is a complex one, and was merely sketched out in section 4, using some illustrative examples. Further work, currently in progress, is developing a comprehensive theory of the role of maptree transformations. In particular, we are developing a classification of kinds of folds, as well as the underlying structure of the labels (a group under concatenation). This forms part of a larger mission, outlined in [9] to develop an event-based approach to dynamic geospatial phenomena.

Acknowledgments. This material is partly based upon work supported by the US National Science Foundation under grant number IIS-0916219.

References

1. Buneman, O.P.: A grammar for the topological analysis of plane figures. In: Meltzer, B., Michie, D. (eds.) *Machine Intelligence*, vol. 5, pp. 383–393. Elsevier (1970)
2. Edmonds, J.R.: A combinatorial representation for polyhedral surfaces. *Notices Amer. Math. Soc.* 7, 646 (1960)
3. Egenhofer, M.J., Franzosa, R.D.: Point-set topological spatial relations. *International Journal of Geographical Information Systems* 5(2), 161–174 (1991)
4. Jiang, J., Nittel, M., Worboys, S.: Qualitative change detection using sensor networks based on connectivity information. *GeoInformatica* 15(2), 305–328 (2011)
5. Jiang, J., Worboys, M.: Event-based topology for dynamic planar areal objects. *International Journal of Geographical Information Science* 23(1), 33–60 (2009)
6. Randell, D.A., Cui, Z., Cohn, A.: A spatial logic based on regions and connection. In: Nebel, B., Rich, C., Swartout, W. (eds.) *KR 1992. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference*, pp. 165–176. Morgan Kaufmann, San Mateo (1992)
7. Stell, J., Worboys, M.: Relations between adjacency trees. *Journal of Theoretical Computer Science* 412, 4452–4468 (2011)
8. Tutte, W.T.: What is a map? In: *New Directions in the Theory of Graphs*, pp. 309–325. Academic Press, New York (1973)
9. Worboys, M.F.: Event-oriented approaches to geographic phenomena. *International Journal of Geographic Information Science* 19(1), 1–28 (2005)

Automatic Creation of Crosswalk for Geospatial Metadata Standard Interoperability

Hui Yang¹ and Gefei Feng^{2,3}

¹ School of Resource and Earth Science, China University of Mining and Technology,
1# Daxue Road, Xuzhou Jiangsu, China
whinee@gmail.com

² Institute of Linguistic, Jiangsu Normal University, 57# Heping Road, Xuzhou Jiangsu, China

³ Jiangsu College Key Lab of Linguistic Sciences and Neuro-cognition Engineering,
57# Heping Road, Xuzhou Jiangsu, China
fspeed@gmail.com

Abstract. Geospatial metadata is very important for describing, managing, querying, retrieving, exchanging and transmitting geospatial data and information resource. As the number, size and complexity of the geospatial metadata standards grow, the task of facilitating greater interoperability between different metadata standards becomes more difficult and important. Crosswalk is the key point to reach interoperability over geospatial metadata standards. Our goal is to provide the automatic creation of crosswalk for heterogeneous geospatial metadata standard interoperability. We introduce a brief but comprehensive overview of the various geospatial metadata standards and describe the related work of geospatial metadata crosswalks. Next, we design a series of formal definitions for geospatial metadata standard mapping. Then, we discuss the multiple attributes similarity of geospatial metadata standard. Next, we introduce the method of automatic creation of crosswalk and mapping based on multiple attribute similarity. Finally we demonstrate our approach and its accuracy using an established crosswalk (CSDGM and ISO 19115).

Keywords: Geospatial metadata standard, interoperability, crosswalk, multi-attribute similarity.

1 Introduction

Geospatial metadata is a type of metadata that captures the content, quality, condition, authorship, and any other basic characteristics of geospatial data and information resource. It is best defined as a formally structured and documented collection of information about geospatial data that represents who produced the geospatial data, what is in them, when they were produced and modified, where the geospatial data originated from, why they were produced, and how the geospatial data can be obtained [1]. The foremost aim of the geospatial metadata standards is to facilitate the ability to describe, manage, query, exchange, transmit, share and integrate geospatial data and information [2].

There are many geospatial metadata standards that have consistently arisen at a regional, national or global level like: The Content Standards for Digital Geospatial Metadata (CSDGM) [3] is the current US Federal Metadata Standard. The FGDC originally adopted the CSDGM in 1994 and revised it in 1998. ISO 19115 "Geographic Information - Metadata" [4] is a standard of the ISO/TC 211 Geographic Information/Geomatics. The committee CEN/TC 287 determines the European standards for geographic information (CEN ISO/TS 19139:2009) [5], and these will be carried out in close co-operation with ISO/TC 211 in order to avoid duplication of work. China's geographic information metadata standards development work is also actively carried out [6], including NREDIS information sharing metadata content standard draft [7], national foundation geographic information system (NFGIS) metadata standard draft [8], etc.

As the number, size and complexity of the geospatial metadata standards grow, the task of facilitating metadata interoperability in different standards becomes more difficult and tedious [9]. The alternative solution is: different organizations still use their own geospatial metadata standards, build mapping relations (Metadata Crosswalks) between two related metadata standards in order to organically combine the heterogeneous geospatial metadata together, which currently is the more commonly-used implementation method to realize geospatial metadata interoperability.

The objective of this paper is to propose an automatic creation of crosswalk approach for geospatial metadata standard interoperability. Based on a series of formal definitions for geospatial metadata standard mapping, the approach is developed through two phases. Firstly, a list of geospatial metadata element name, description and structure, is discussed as multiple attributes similarity. Secondly, the method of automatic creation of crosswalk is performed in order to discover the mapping relationships between two heterogeneous geospatial metadata standards.

The remainder of this paper is structured as follows. First we introduce related work on geospatial metadata crosswalks and standard mapping. Then, we discuss the multiple attributes similarity of geospatial metadata standard. Next, we introduce the method of automatic creation of crosswalk and mapping. We then demonstrate our approach and its accuracy using an established crosswalk (CSDGM and ISO 19115). Finally, the conclusions are given.

2 Related Work

This section introduces related work on geospatial metadata crosswalks and provides a series of formal definitions for geospatial metadata standard mapping.

2.1 Geospatial Metadata Crosswalks

Crosswalk is often-used approach for geospatial metadata interoperability. A crosswalk is a table that maps the relationships and equivalencies between two or more metadata formats. Crosswalks or metadata mapping support the ability of search

engines to search effectively across heterogeneous database, i.e. crosswalks help promote interoperability [10].

Currently some geospatial metadata provide many crosswalk technologies, which is usually made into tables, namely the metadata mapping table, or metadata mapping dictionary. Metadata Architecture and Application Team (MAAT) has collected international metadata standards and provided mapping for different metadata elements [11]. FGDC has conducted the study of the mapping between two kinds of metadata standards, CSDGM and ISO 19115, and the published FGDC CSDGM to ISO 19115 Crosswalk [12] is the specific embodiment of the mapping relation. The Alexandria digital library provides FGDC to USMARC and ADL to FGDC [13] crosswalks.

The international cooperation project--Dublin Core Metadata Initiative from Ohio University Library Center (OCLC) [14] has launched a Web service that solves the problem of metadata interoperability in the network environment: metadata mapping service (CWS, Crosswalk Web Services). The service employs technology of XML and XSLT to transfer any two metadata Schema and the prototype system supplied provides a flexible environment for the transformation of metadata model [15]. However, the grammar and semantic of the metadata is completely separated in this approach, which is also the limitation of this method; therefore, the multiple information that metadata contains should be considered in a more comprehensive way so as to proceed multi-type automatic mapping discovery. Tang, etc put forward ontology mapping method called RiMOM [16] based on Bayesian decision theory, which formalizes the ontology mapping problem to risk decision making and proposed multi-strategy mapping discovery mechanism with the combination with a variety of metadata information. However, the method fails to take the characteristics of geographic information into consideration, which makes it inappropriate and impossible to be directly applied into the automatic creation of metadata crosswalk.

2.2 Formal Definitions of Geospatial Metadata Standard Mapping

The geospatial metadata standard is a collection of hierarchically organized elements that define the content of metadata record. These elements fall into one of three categories – sections, compound elements or data elements. Each section begins with the name and definition of the compound elements. The compound element is composed of a group of other compound elements and data elements. The data element is a logically primitive item of geospatial metadata. They simply describe the relationship among other elements giving an overall structure to the geospatial metadata standard.

Geospatial Metadata standard can be defined as a 4-tuple

$$M = \{S, C, D, H\} \quad (1)$$

Where,

- S is the set of sections,
- C is the set of compound elements,
- D is the set of data elements,
- H is the set of hierarchical structure, $H \subseteq (S \cup C \cup D) \times (S \cup C \cup D)$.

An element $e(e \in (S \cup C \cup D))$ is the logically primitive item of geospatial metadata standard, which can be one of three possible types: data element, compound element, or a section. In a section, compound elements can be defined by data elements or other compound elements. Compound elements represent higher-level concepts that cannot be represented by individual data elements.

To handle the hierarchical structure of section, compound element and data element, a conventional 4-tuple is not expressive enough. Thus, we define the hierarchical tree structure with a set of linked elements, which are the extension definitions of geospatial metadata standard.

Therefore, the hierarchical relation between elements can be defined as a 2-tuple $(e_i, e_j) \in H$, where e_i is the sub-element of e_j . Each element in geospatial metadata standard has zero or more child elements. An element that has a child is called the child's parent elements. An element has at most one parent. The elements with the same parent are called sibling elements.

Given $e \in (S \cup C)$, and the **parent element** is defined as $P(e)$

$$P(e) = \{e_i \mid e_i \in (S \cup C) \wedge (e, e_i) \in H\} \quad (2)$$

Given $e \in (C \cup D)$, and the **child element** is defined as $C(e)$

$$C(e) = \{e_i \mid e_i \in (C \cup D) \wedge (e_i, e) \in H\} \quad (3)$$

Given $e \in (S \cup C \cup D)$, and the **sibling element** is defined as $B(e)$

$$B(e) = \{e_i \mid \forall p \in (S \cup C), e_i \in (S \cup C \cup D) \wedge (e, p) \in H \wedge (e_i, p) \in H \wedge e_i \neq e\} \quad (4)$$

The **mapping** of geospatial metadata standards is "lateral" mapping from one standard to another.

$$Map : M_1 \rightarrow M_2 \quad (5)$$

Where, M_1 is the source geospatial metadata standard, M_2 is the target geospatial metadata standard, and Map means the mapping from M_1 into M_2 .

The **crosswalk** is to make a comparison between the elements of two heterogeneous geospatial metadata standards and to build relative conceptual and structural association for elements in the two standards. The crosswalk between geospatial metadata standards is a mapping function, which can also be represented by:

$$Map(\{e_1\}) = \{e_2\} \quad (6)$$

The crosswalk is a table that shows equivalent in more than one geospatial metadata standard. It maps the elements $\{e_1\}$ in M_1 to the equivalent elements $\{e_2\}$ in another geospatial metadata standard M_2 .

3 Automatic Creation of Crosswalk

On the basis of the formal definitions of geospatial metadata standard, it is possible to point out multi-attribute similarity between elements, such as element name, description and structure. Based on these conceptual and structural attributes, the calculation method of similarity is defined. Finally, the automatic mapping discovering method that synthesizes multi-attribute similarity value is presented. An overview of this process is given in Figure 1.

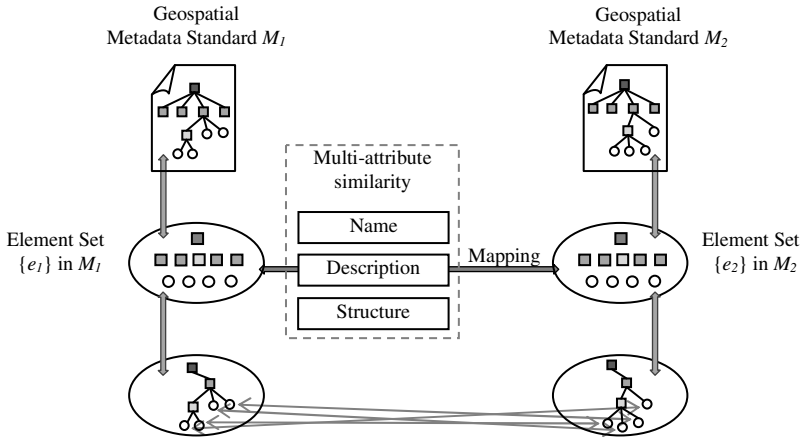


Fig. 1. Automatic creation of crosswalk process

3.1 Multi-attribute Similarity

In order to know what to consider when mapping two elements of different geospatial metadata standards, in this section we define three kinds of attribute similarity that should be taken into account when comparing metadata elements, both at the conceptual-level and at the structural-level respectively.

Element Names

Name is the label for the metadata element. It is easily comprehended and unambiguous. The most intuitive method maybe that of exploiting element names to discover the mapping. The name of an element usually contains one or several words. Therefore, before calculating the similarity value of the two names, the segmentation of words is the first priority to do. For example, if the name of a metadata element is “Time Period of Content”, after separating, we can get {Time, Period, of, Content}. Through this pretreatment, the name of an element can be simply viewed as a set of multiple words and the similarity of names can be defined as name similarity sim_n between elements e_1 and e_2 as:

$$sim_s(e_1, e_2) = sim(e_1.name, e_2.name) = sim(name_1, name_2) \tag{7}$$

Where, $sim_n(e_1, e_2)$ denotes the name similarity between elements e_1 and e_2 , $name_1 = \{w_{1i} | i=1,2,\dots,n\}$ and $name_2 = \{w_{2j} | j=1,2,\dots,n\}$ is the acquired word set after segmentation, which can be processed into two token sets as $\{w_{1i}\}$ and $\{w_{2j}\}$.

Then the similarity matrix of all the words that the two element names contain is calculated. Each value in the matrix means the similarity of some two words in the elements names after scanning, comparing and calculating character by character. However, two element names with similar sense might be absolutely differently spelled. E.g. “Browse Graphic” in FGDC is equivalent to “Graphic Overview” in ISO. So for every word w_{1i} in $name_1$, a word that shares the maximum word similarity with $sim_n(w_{1i}, w_{2j})$ is chosen from $name_2$ as the similarity between w_{1i} and $name_2$, generating $sim_s(w_{1i}, name_2) = \max(sim_n(w_{1i}, w_{2j}))$. And the name similarity is finally defined to be:

$$sim_s(name_1, name_2) = \frac{1}{n} \sum_{i=1}^n sim(w_{1i}, name_2) = \frac{1}{n} \sum_{i=1}^n \max(sim(w_{1i}, w_{2j})) \quad (8)$$

Where, n is the number of words in $name_1$. The similarity value of two words is calculated by combining conceptual similarity and statistical similarity:

$$sim_n(w_{1i}, w_{2j}) = \frac{sim_c(w_{1i}, w_{2j}) + sim_s(w_{1i}, w_{2j})}{2} \quad (9)$$

Where, $sim_c(w_{1i}, w_{2j})$ denotes the conceptual similarity between w_{1i} and w_{2j} according to the thesaurus. $sim_s(w_{1i}, w_{2j})$ is the statistical similarity which will be described later.

With the rapid development of language processing technology, many thesauruses such as EuroWordNet [17], HowNet [18], and Russian WordNet [19] have been developed with reference to WordNet [20] which is developed by Princeton University. WordNet has a semantic network of word senses, in which each semantic node is a synset, represents a set of synonymous words.

That is to say, a semantic node contains words with same sense and a word can occur in different nodes indicating that the word has multiple senses. According to WordNet, Lin defines the conceptual similarity between two words as the maximum semantic similarity between their senses [21]:

$$sim_c(w_{1i}, w_{2j}) = \max(sim_c(s_1, s_2)) = \frac{2 \times \log P(s)}{\log P(s_1) + \log P(s_2)} \quad (10)$$

Where, s_1 and s_2 is the semantic nodes of w_{1i} and w_{2j} respectively. Semantic node S is the shared parent node of s_1 and s_2 . $P(s) = count(s)/total$, is the probability of a randomly selected word occurring in semantic nodes S or any semantic nodes

below it. $total$ denotes the number of words in WordNet, and $count(s)$ is the word count in s and its child nodes.

Based on statistical method, Pantel and Lin [22] build an electronic dictionary, in which similarity between two words is defined to be their distribution in a corpus. The statistical similarity of two words $sim_s(w_{1i}, w_{2j})$ can be acquired by directly looking up the dictionary.

Element Description

The standardization organizations usually provide their own corresponding terms or glossary table in the development of geospatial metadata standards, and these tables is the set of a variety of words and descriptor of geospatial metadata. Element descriptor refers to the text description information of elements that exist in form of natural language, which is also important mapping discovery information that can help find mapping relation between the elements that have different names but share some semantic associations.

The descriptor of each element can be seen as a “text”. The mapping between elements can be found by using the word and its frequency information appeared in the “text”, thus the problem is transformed into a text classification problem. As for the two given geospatial metadata M_1 and M_2 , $D_{1i} = \{d_{1i}\}$, $D_{2j} = \{d_{2j}\}$ are the descriptor set of element e_{1i} and e_{2j} respectively. In text classification method, the descriptor of M_2 is viewed as training sample and that of M_1 is test sample. The mapping discovery is reached by predicting the classification of the test sample.

There are many ways to realize text classification. In this paper, Naive Bayesian classifier (NB) [23] is adopted. NB learns classification model from training text and then choose the maximum posterior probability $\arg_{e_{2j}} \max(p(e_{2j}|D_{1i}))$ to predict the category of descriptor D_{1i} (corresponds to e_{2j} in M_2). That is, to all the possible candidate mapping of e_{1i} , this method can calculate the predictive value of this mapping and the maximum predictive value is taken as the description similarity sim_d .

$$sim_d(e_1, e_2) = \arg_{e_{2j}} \max(p(e_{2j}|D_{1i})) \quad (11)$$

Where, the posterior probability $p(e_{2j}|D_{1i})$ is defined to be:

$$p(e_{2j}|D_{1i}) = \frac{p(D_{1i}|e_{2j})p(e_{2j})}{p(D_{1i})} \quad (12)$$

Where, $p(D_{1i})$ is a normalization constant, which can be neglected. $p(e_{2j})$ is the proportion that the number of training instances takes in e_{2j} . For the given e_{2j} , with the hypothesis that the distribution of words appeared in the description D_{1i} are

independent of each other. Thus we can rewrite the equation by define the $p(D_{li}|e_{2j})$ as $p(D_{li}|e_{2j}) = \prod_{w \in D_{li}} p(w|e_{2j})$.

$$p(e_{2j}|D_{li}) = \prod_{w \in D_{li}} p(w|e_{2j})p(e_{2j}) \tag{13}$$

In the equation, $p(w|e_{2j}) = n(w, e_{2j})/n(e_{2j})$. $n(e_{2j})$ is the total number of words appeared in all the descriptions of e_{2j} , and $n(w, e_{2j})$ is the number of times that word w appears in the description of e_{2j} .

Element Structure

Element structure mapping describes the structural similarity between two elements. If two given elements have the same or similar contextual structure, then they may be matchable. For example, if the parent and child elements of the two elements have mapping relation respectively, then the possibility that the two elements have mapping will be greater. That is to say, two parent elements match if their child elements match.

The contextual structure of elements includes its parent, child and sibling elements. Thus, the structural similarity sim_s of two elements can be defined to be the average similarity of structural elements in their contexts. At present this paper only takes immediately associated elements such as parent, child and sibling elements as its contextual elements.

$$sim_s(e_1, e_2) = \frac{sim(P(e_1), P(e_2)) + sim(C(e_1), C(e_2)) + sim(B(e_1), B(e_2))}{3} \tag{14}$$

Where, $sim(P(e_1), P(e_2))$, $sim(C(e_1), C(e_2))$, and $sim(B(e_1), B(e_2))$ is the similarity between parent elements, child elements, and sibling elements of e_1 and e_2 respectively. The similarities of the contextual elements can be acquired by calculating element name or description similarity.

3.2 Mapping Discovery Based on Multi-attribute Similarity

After the determination of multiple attribute similarity, the immediate approach is to determine the weight of each attribute similarity. That is, we need to estimate the relative importance granted to each attribute similarity. This paper first constructs pairwise comparison matrix, then calculate the weight of each attribute similarity by adopting sum value method which possesses high reliability and small error. The process of weight determination is as follows:

The first step is the comparison of the multi-attribute similarity. The multi-attribute similarities are compared pairwise according to their importance of influence and based on the standardized comparison scale. The result of the pairwise comparison on multi-attribute similarity can be summarized in an evaluation matrix S in which every element s_{ij} is the quotient of weights of the attribute similarity, as shown below.

$$S = \begin{bmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{bmatrix} = \begin{bmatrix} \frac{sim_n}{sim_n} & \frac{sim_d}{sim_n} & \frac{sim_s}{sim_n} \\ \frac{sim_n}{sim_d} & \frac{sim_n}{sim_d} & \frac{sim_s}{sim_d} \\ \frac{sim_d}{sim_d} & \frac{sim_d}{sim_d} & \frac{sim_s}{sim_d} \\ \frac{sim_n}{sim_s} & \frac{sim_d}{sim_s} & \frac{sim_s}{sim_s} \\ \frac{sim_n}{sim_s} & \frac{sim_s}{sim_s} & \frac{sim_s}{sim_s} \end{bmatrix} = \begin{bmatrix} 1 & \frac{sim_d}{sim_n} & \frac{sim_s}{sim_n} \\ \frac{sim_n}{sim_d} & 1 & \frac{sim_s}{sim_d} \\ \frac{sim_n}{sim_s} & \frac{sim_d}{sim_s} & 1 \end{bmatrix} \quad (15)$$

Where, for the selected three multi-attribute similarity sim_n , sim_d and sim_s , the comparison matrix S will be 3×3 and the entry s_{ij} will denote the relative importance of attribute similarity i with respect to the attribute similarity j . In the matrix $S = (s_{ij})_{3 \times 3}$, $S_{ii} = 1$ if when $i = j$ and $S_{ij} \times S_{ji} = 1$.

Specific to the reciprocal comparison matrix S , specification column sum method, latent root value method (geometric method) and characteristic root method are the more used method to determine the weight of the attributes. Specification column method is adopted in this paper in allusion to the study content: first, normalize the rows, and then calculate the column average [24].

$$w = (w_1, w_2, w_3)^T = \left[\frac{1}{3} \sum_{j=1}^3 \frac{S_{1j}}{\sum_{k=1}^3 S_{kj}}, \frac{1}{3} \sum_{j=1}^3 \frac{S_{2j}}{\sum_{k=1}^3 S_{kj}}, \frac{1}{3} \sum_{j=1}^3 \frac{S_{3j}}{\sum_{k=1}^3 S_{kj}} \right]^T$$

, and $\sum_{i=1}^n w_i = 1$ (16)

Where, w_1 , w_2 and w_3 are the weight of multi-attribute similarity sim_s , sim_d and sim_n respectively. Last but not least, the weighted sum function is chosen as the evaluation function of multiple-attribute synthetic similarity S_i , the bigger the function value is, the more rational the corresponding candidate mapping is.

$$S_i = w_1 \times sim_n + w_2 \times sim_d + w_3 \times sim_s \quad (17)$$

There is no two geospatial metadata standards that are 100% equivalent. One standard may have an element that doesn't exist in another standard, or it may have an element that is split into two or more different elements in another standard. Therefore, there are six types of mapping among elements between two different geospatial, including one-to-one, one-to-null, null-to-one, one-to-many, many-to-one and many-to-many.

One-to-one mapping is the simplest and most frequently-used mapping type, whose mapping discovery can be realized by choosing the mapping element that the

maximum of all the similarities (name sim_n , description sim_d , structure sim_s , and synthesized similarity) correspond to respectively for every element in M_1 .

One-to-null mapping means as for every element in M_1 , there is no direct corresponding element in M_2 , nor corresponding element combination. One-to-null is a special mapping, which can be discovered through the following heuristic rule: (1) As for e_{1i} , if the maximum name similarity or structure similarity is smaller than the smallest given threshold value. (2) For one element e_{1i} in M_1 , if both the parent element $e_{1i}^p = P(e_{1i})$ and child element $e_{1i}^c = C(e_{1i})$ have the mapped target element e_{2i}^p and e_{2i}^c and there is no other element between e_{2i}^p and e_{2i}^c , that is, $e_{2i}^p = P(e_{2i}^c)$ or $e_{2i}^c = C(e_{2i}^p)$.

As for the discovery of one-to-many mapping, when there is no direct element in M_2 that correspond to certain element in M_1 , the next step is to search the entire mapping space; If multiple target elements are mapped into the same source element, then the combined target element set are to represent a mapping. When the concept element in source metadata standard is rough and broad, and element definition in the target metadata standard elements is more detailed and clear, that is, the discovery of one-to-many mapping only exist when the semanteme that one certain element in source metadata standards expresses may include the semanteme that multiple elements in the target metadata standards express.

Null-to-one can be viewed as the anti-type of one-to-null, that is, the mapping of null-to-one can be found through reverse mapping. But once we have mapped many-to-one we can't map them back into one-to-many. So separate crosswalk would be required to map from standard M_1 to M_2 and from standard M_2 to M_1 [25]. While the discovery of one-to-many, many-to-one, and many-to-many mapping is a very complex issue, which will not be considered as a priority in this paper.

4 Experimental Results

Two types of metadata mapping discovery, namely one-to-one and one-to-null were conducted on the geospatial metadata standard data set (CSDGM) of FGDC and geographic information metadata standard (ISO 19115) of ISO/TC 211. The statistical data of two experimental data sets is as shown in table 1.

Table 1.Statistic data for element sets of CSDGM and ISO 19115

geographic information metadata standard	Element	Element section	Compound element	Simple element
CSDGM	351	10	123	218
ISO19115	419	10	95	314

Two kinds of mapping (one-to-one and one-to-null) assessment methods based on name, descriptor, and contextual structure of elements and automatic mapping discovery were conducted on the two geospatial metadata element sets. Element name-based, element descriptor-based and element structure-based single-attribute decision-making was chosen as a baseline method for validating mapping performance of the method of automatic mapping discovery based on multi-attribute similarity. The mapping that FGDC-released FGDC CSDGM to ISO 19115 Crosswalk expresses was taken as the evaluation criteria from this mapping discovery software is as shown in Figure 2.

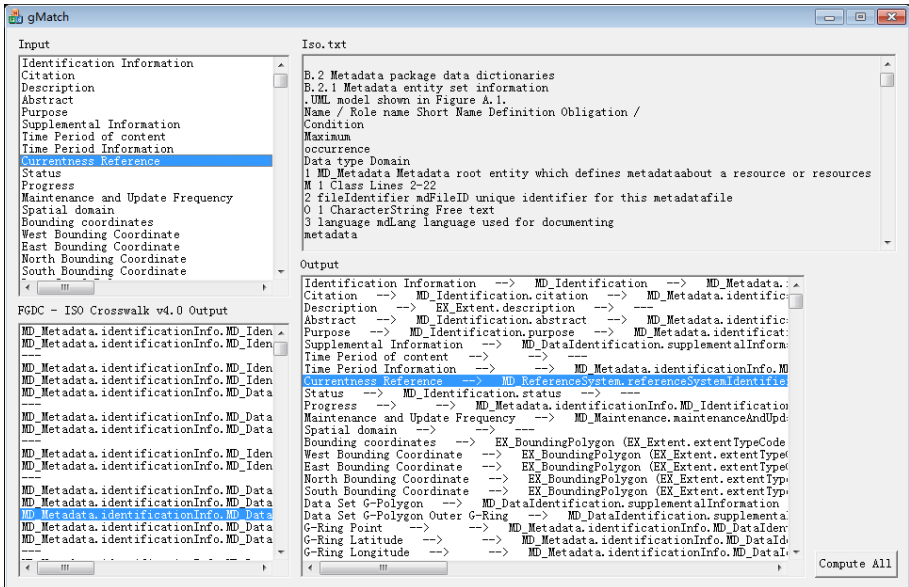


Fig. 2. Geospatial metadata automatic mapping discovery software

Table 2 is the comparative analysis of mapping results obtained from the four decision-making criteria and that of Crosswalk respectively; precision and recall were adopted to assess experimental results. The precision is the percentage of correct discovered mapping, and the recall is the percentage of the discovered mapping.

Table 2. Contrast of experience results

Criteria	one-to-one		one-to-null	
	Precision	Recall	Precision	Recall
Element name	73.34%	76.17%	71.93%	78.71%
Element description	54.58%	61.46%	59.31%	66.28%
Element structure	76.45%	79.07%	74.57%	79.30%
Multi-attribute similarity	87.83%	90.89%	86.35%	88.17%

The experimental result shows:

1. Preferable mapping result. In the mapping of Geographic information metadata standard data set of CSDGM and ISO19115, one-to-one mapping: the precision is 87.83%, the recall is 90.89%; one-to-null mapping: the precision is 86.35%, the recall is 88.17%. The results show that the proposed automatic metadata mapping method is effective; in the experimental mapping tasks of one-to-one and one-to-null, this method can achieve good results and the average precision and recall of the two are 87.09 % and 89.53%.

2. Superior to the result from three single-attribute methods. Compared with the element name-based method, automatic mapping model based on multi-attribute group decision-making improved the mapping results; the precision of one-to-one mapping increased 14.49%, the recall is 14.72%; while the precision of one-to-null mapping put up 14.42%, the recall 9.46%. The improvement of the mapping effect is more obvious compared with element description-based method: for one-to-one mapping, the precision increased 33.25%, the recall 29.43%; for one-to-null mapping, the precision reached an increase of 27.04%, the recall an increase of 21.89%. As for the comparison with single-attribute decision-making based on elements structure, the mapping result was also improved. For one-to-one mapping, the precision realized a rising of 11.38%, recall of 11.82%; while for one-to-null mapping, the precision achieved a climbing of 11.78%, the recall of 8.87%. The above data shows that Multi-attribute group decision making method is better than three types of baseline methods with an average precision and recall improvement of 18.73% and 16.03% respectively.

3. The experimental result of mapping one-to-many and many-to-one is not provided. CSDGM, ISO19115 and FGDC CSDGM to ISO 19115 Crosswalk are developed by the International Organization for Standardization, ensuring good representation and favorable evaluation reference value, which is taken into consideration in the selection of experiment result. The elements definition of CSDGM and ISO19115 are relatively detailed and clear, providing no one-to-many mapping, which also shows that the application of automatic mapping method based on multi-attribute similarities is in need of more experiments data set in the discovery of complex mapping to realize supplement and perfection of the experiment analysis and evaluation.

5 Conclusions

Based on multi-attribute similarity, automatic creation of crosswalk for geospatial metadata standard interoperability is put forward in this paper. With the utilization of all sorts of information of geographic information metadata, multi-attribute similarities based on element name, description and structure is implemented, which can support the two types mapping discovery of one-to-one and one-to-null. In the comparison with FGDC CSDGM to ISO 19115 Crosswalk mapping table completed by FGDC, the experiments show that the automatic creation of crosswalk for geospatial metadata standards interoperability based on multi-attribute similarities has

good precision and recall. It is superior to single-attribute similarity method based on elements name, description and structure. Therefore, the study will promote effective management and interoperation of heterogeneous geospatial metadata and facilitate the in-depth study of geographic information sharing.

Acknowledgements. The authors would like to thank the financial support by the National Natural Science Foundation of China (Grant No. 41001230), the China Postdoctoral Science Foundation (Grant No. 2012M510193) and Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD). We would like to thank our three anonymous reviewers for their very helpful comments. Their constructive feedback greatly improved our paper.

References

1. Yeung, A., Hall, G.: Spatial Database Systems. GeoJournal Library, vol. 87(pt. 2), pp. 129–173. Springer, Heidelberg (2007)
2. Yang, H., Sheng, Y.H., Wen, Y.N., Hu, Y.: Distributed Geographic Models Sharing Method Based on Web Services. Geomatics and Information Science of Wuhan University 34(2), 142–145 (2009)
3. FGDC. Content Standard for Digital Geospatial Metadata, <http://www.fgdc.gov/metadata/geospatial-metadata-standards>
4. ISO. ISO 19115:2003, Geographic information–Metadata, <http://www.isotc211.org/>
5. CEN/TC 287, Geographic Information-Data description-Metadata, <http://www.centc287.eu/index.php/standards>
6. Liu, R.M., Jiang, J.T.: Implementation of Metadata Standard of Information Sharing for Sustainable Development of China. Research on the China Geographic Information Metadata Standard. Sciences Publishing House (1999)
7. National Geospatial Information Exchanging Center. Standard Draft of Data-sharing Metadata Content of NREDIS (2000), <ftp://ftp.nsii.gov.cn/pub/standard/standard02.zip>
8. National Geomatics Center of China. Metadata Standard Draft of National Fundamental Geographic Information System (First Draft), <http://nfgis.nsdi.gov.cn/nfgis/chinese/bz/mt0.htm>
9. Noguera-Iso, J., Zarazaga-Soria, F.J., Lacasta, J., Bejar, R., Muro-Medrano, P.R.: Metadata standard interoperability: application in the geographic information domain. Computers, Environment and Urban Systems 28(6), 611–634 (2004)
10. DCMI. Dublin Core Metadata glossary, Final draft, <http://dublincore.org/documents/2001/04/12/usageguide/glossary.shtml>
11. Metadata Architecture and Application Team (MAAT), http://metadata.teldap.tw/standard/mapping-foreign_eng.html
12. FGDC. FGDC CSDGM to ISO 19115 Crosswalk, http://www.fgdc.gov/metadata/documents/FGDC_Sections_v40.xls/view
13. FGDC to USMARC, <http://www.alexandria.ucsb.edu/public-documents/metadata/fgdc2marc.html>
14. HowNet, <http://www.keenage.com>
15. Godby, C.J., Smith, D., Childress, E.: Two Paths to Interoperable Metadata. In: DC 2003: Supporting Communities of Discourse and Practice-Metadata Research & Applications, Seattle, Washington (2003)

16. Tang, J., Li, J., Liang, B., Huang, X., Li, Y., Wang, K.: Using Bayesian Decision for Ontology Mapping. *Web Semantics: Science, Services and Agents on the World Wide Web* 4(4), 243–262 (2006)
17. Vossens, P.: EuroWordNet: a multilingual database for information retrieval, <http://dare.uvu.nl/bitstream/1871/11136/1/Delos97.pdf>
18. HowNets, <http://www.keenage.com>
19. Loukachevitch, N.V.: Russian language in cross-language information retrieval, <http://www.clef-campaign.org/workshop2003/presentations/loukachevitch.ppt>
20. Princeton University Cognitive Science Laboratory, WordNet, <http://wordnet.princeton.edu/>
21. Lin, D.: Using syntactic dependency as local context to resolve word sense ambiguity. In: *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics Madrid, Spain*, pp. 64–71 (1997)
22. Pantel, P., Lin, D.: Discovering word sense from text. In: *Proceedings of the 2002 ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada*, pp. 613–619 (2002)
23. Mitchell, T.M.: *Machine Learning*. McGraw Hill, Columbus (1997)
24. Saaty, T.L.: *The Analytic Hierarchy Process*. McGraw Hill, New York (1980)
25. Caplan, P.: *Metadata Fundamentals for All Librarians*. American Library Association, Chicago (2003)

A Dartboard Network Cut Based Approach to Evacuation Route Planning: A Summary of Results

KwangSoo Yang*, Venkata M.V. Gunturi, and Shashi Shekhar

Department of Computer Science, University of Minnesota, Minneapolis, MN 55455
{ksyang,gunturi,shekhar}@cs.umn.edu

Abstract. Given a transportation network, a population, and a set of destinations, the goal of evacuation route planning is to produce routes that minimize the evacuation time for the population. Evacuation planning is essential for ensuring public safety in the wake of man-made or natural disasters (e.g., terrorist acts, hurricanes, and nuclear accidents). The problem is challenging because of the large size of network data, the large number of evacuees, and the need to account for capacity constraints in the road network. Promising methods that incorporate capacity constraints into route planning have been developed but new insights are needed to reduce the high computational costs incurred by these methods with large-scale networks. In this paper, we propose a novel scalable approach that explicitly exploits the spatial structure of road networks to minimize the computational time. Our new approach accelerates the routing algorithm by partitioning the network using dartboard network-cuts and groups node-independent shortest routes to reduce the number of search iterations. Experimental results using a Minneapolis, MN road network demonstrate that the proposed approach outperforms prior work for CCRP computation by orders of magnitude.

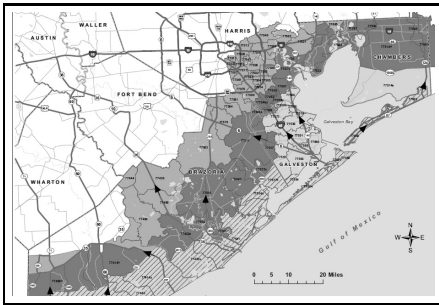
Keywords: evacuation route planning, spatial network, dartboard network cut, routing and scheduling algorithm.

1 Introduction

Hurricane Rita and the recent Tohoku tsunami that hit Japan are reminders that evacuation planning is an essential component of civic emergency preparedness. One of the most important requirements of evacuation planning is to protect population during a disaster. Ensuring the safety of all residents of a structure, city, or region during a disaster requires evacuation planning tools to produce the safest and most efficient route schedules for large scale road networks and populations within limited time constraints. Consider a hurricane evacuation planning problem. Low lying riverside and coastal regions, are especially at risk for a major storm or flooding as shown in Figure [1\(a\)](#). The speed and direction

* Corresponding author.

of a hurricane can change rapidly, so the threat to particular areas of the coast may come up suddenly. Massive emergency evacuation from these areas brings more challenges for civic authorities due to the large and unpredictable shape of evacuation zones (EZs) along coastal areas. In 2005, the approach of hurricane Rita provoked one of the largest evacuations in U.S. history, resulting in three million evacuees. During the evacuation, the enormous number of people fleeing from the Houston area coupled with a number of shortcomings in exit routes for residents caused massive traffic jams. In 1992, Hurricane Andrew, the third most powerful storm to hit the Florida coast caused massive delays and major congestion [1].



(a) Houston Hurricane EZs. Courtesy: <http://www.hcoem.org>



(b) Congestion From Rita on I-45. Courtesy: FEMA

Fig. 1. Houston EZ and Congestion from the hurricane Rita

Previously, disasters like Rita and Andrew demonstrated the inadequacy of hand drawn plans for evacuating populations after a disaster. They also demonstrated the need to account for the capacity constraints of road networks. Computational methods of evacuation planning promise more efficient route schedules in the face of massive storms. These methods must be scalable and able to produce results easily in a short time frame. Furthermore, they must be able to handle dynamic environments.

Over the last two decades there has been a considerable amount of research on route planning for evacuation zones and other event scenarios. Recent work on evacuation route planning can be divided into three categories: (1) Linear Programming (LP) methods that use a network flow problem to minimize the total evacuation time [4,17,11,10], (2) Simulation methods that model the evacuation route as individual movements [5,10] or a traffic assignment problem [16], and (3) Heuristic methods that use an approximate optimization technique to minimize the computation time. The LP approach uses iterative algorithms (e.g., simplex or ellipsoid method) to minimize the cost function based on given constraints [4,17,11]. This approach requires using a static network model for a dynamic environment to generate optimal evacuation plans. Consequently, the transportation network needs to be transformed into a time-expanded graph (TEG) by constructing $T + 1$ copies of nodes and edges [8]. Unfortunately, the

number of variables and iterations in this linear program is in general exponential in the size of the underlying network, limiting its usefulness for large scale networks [7]. Simulation methods use individual traveler behaviors or traffic assignment in greater detail including the interaction between vehicles. One problem with this model is that regulating individual movements or assigning traffic flow associated with Wardrop’s equilibrium model [22] in emergency evacuation is very complicated, making it inappropriate for large evacuation scenarios. Finally, heuristic methods can be used to incorporate capacity constraints into route planning and find near-optimal evacuation plans with reduced computational cost. This is useful for medium-size transportation networks within a limited amount of time. A well known approach for this category is the Capacity Constrained Route Planner (CCRP) [24][15]. However, CCRP incurs excessive computational cost for large network and population datasets.

Recent approaches may not be able to scale up to large size transportation networks on densely populated regions due to the limited capacity constraints of road networks and large numbers of evacuees. New insights are needed to reduce the high computational costs where these methods incur with large-scale networks. Our work focuses on minimizing computation time and enhancing scalability for large transportation networks. We explore a novel routing algorithm that exploits the underlying spatial network structure of road networks. A common evacuation scenario displays dartboard network structure leading to be partitioned by dartboard network cuts (DBN-cuts). We introduce the notion of dartboard network structure and explain how to organize and group evacuation routes. Our new approach accelerates the routing algorithm by grouping multiple node-independent shortest routes to reduce the number of search iterations. For example, instead of a single-route shortest-path algorithm, we use a node-independent shortest-paths algorithm to aggregate evacuees on different spatial locations and evaluate evacuation routes without sacrificing the quality of the evacuation route plan.

Our Contributions: In this paper, we propose a novel algorithm that uses an underlying dartboard network structure driven by DBN-cuts. DBN-cuts group multiple node-independent shortest routes and reduce iterations of an evacuation routing algorithm. We use a generalized node-independent shortest path algorithm to obtain these node-independent routes and minimize the computational time. Specifically, our contributions are as follows:

- We propose a dartboard network structure based on common evacuation scenarios.
- We propose a dartboard network-cut for evacuation route planning (DBNC-ERP) algorithm to group multiple node-independent routes based on DBN-cuts.
- We provide a cost model for the DBNC-ERP.
- We experimentally evaluate the proposed algorithm and validate the cost model using real road network datasets.

Scope and Outline: This paper proposes a novel evacuation route planning algorithm for large scale network datasets based on dartboard network structure. Our approach uses node-independent shortest routes for DBNC-ERP. The rest of the paper is organized as follows: Section 2 provides the problem definition. Section 3 presents our proposed approach. In Section 4, we discuss how our cost model to predict the performance could be derived. Section 5 describes the experiment design and presents the experimental observations and results. Finally, Section 6 concludes the paper.

2 Problem Definition

The problem of evacuation route planning can be formalized as follows: Given a transportation network with maximum node and edge capacity constraints, initial node occupancy, and destination locations, our objective is to find evacuation route scheduling that minimizes the evacuation time and minimizes the computational cost. We formally define the problem as follows:

Input: A transportation network with

- non-negative integer capacity constraints on nodes N and edges E ,
- the total number of evacuees and their initial location, and
- location of evacuation destination

Output: An evacuation plan consisting of a set of origin-destination routes and a scheduling of evacuees on each route.

Objective:

- Minimize the computational cost of producing the evacuation plan
- Minimize the evacuation time.

Constraints:

- Edge travel time preserves FIFO (First-In First-Out) property.
- The scheduling of evacuees on each route observes the capacity constraints.
- Limited amount of computer memory

3 Dartboard Network Cuts for Evacuation Route Planning

Basic Concept: Spatial networks are represented and analyzed as a graph composed of nodes N and edges E . Every node N represents spatial location in geographic space with a number of evacuees and a node capacity. Every edge E represents a connection between two nodes and has a travel time with an edge capacity. A sequence of nodes $n_1, n_2, n_3, \dots, n_n$ is called a path (or route) if there is an edge between two consecutive nodes. A tree is an undirected graph in which any two nodes are connected by exactly one simple path. A forest is a disjoint union of trees. A set of paths from a source node s to a destination node d is node-independent path if none of the paths share any nodes aside from s and d . In Figure 2(a), for example, there are two node-independent paths traversing from

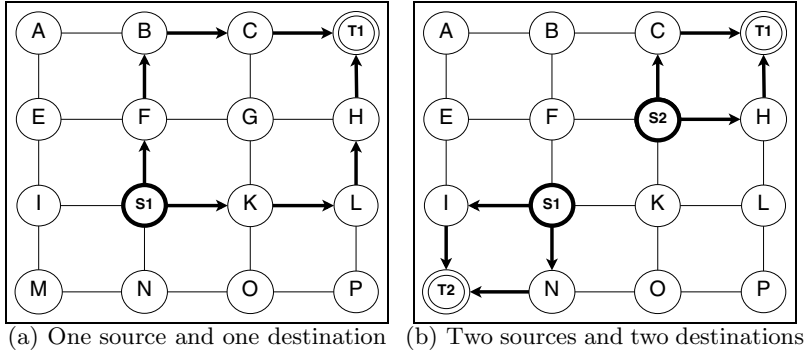


Fig. 2. Node-independent routes in a grid-like network

$S1$ to $T1$ ($S1 \rightarrow F \rightarrow B \rightarrow C \rightarrow T1, S1 \rightarrow K \rightarrow L \rightarrow H \rightarrow T1$). Figure 2(b) shows four node-independent paths based on two pairs of source and destination nodes ($S1 \rightarrow I \rightarrow T2, S1 \rightarrow N \rightarrow T2, S2 \rightarrow C \rightarrow T1, S2 \rightarrow H \rightarrow T1$). These node-independent paths are not necessarily unique so that there may be more than one way of choosing a set of independent paths.

Theorem 1. Given a pair of nodes u, v , the upper bound of the number of node-independent paths is $\min(\text{the degree of a node } u, \text{ the degree of a node } v)$.

Proof. Let m be $\text{degree}(u)$ and n be $\text{degree}(v)$. First, assume that $m \geq n$ and there exist n independent paths. When we add one more independent path, no incoming edge of n exists to obtain the independent path. Second, assume that $m < n$ and there exists m independent paths. Again, we cannot add one more independent path because there is no available outgoing edge of m . Consequently, the maximum of node-independent paths is bounded by $\min(\text{degree}(u), \text{degree}(v))$.

Why are node-independent routes important for capacity constrained route planning algorithms? The key idea is that node-independent routes never share each other’s capacity constraints at the same time during the route evaluation process.

3.1 Dartboard Network Structure

Spatial road networks were shaped in response to socioeconomic activities maximizing ease of navigation in the areas. The structures are neither trees nor perfect grids, but a combination of these structures that emerges from the social and constructive processes. The networks may be broken down into independent routes: most simply, routes that do not share any local parameters, such as node and edge capacity. These independent routes can partition evacuees and use discreet flows to compute the evacuation routes. Consider, for example, the grid-like road network in Figure 2. Because independent routes do not share the capacity constraints of other roads in the network, one node-independent shortest path algorithm for these independent routes can minimize the computational

time. Unfortunately, many road networks have low degree intersections, making it hard to retrieve many node-independent routes. It is known that the mean degree of intersections in the US interstate road network is only about 2.86 [9]. By Theorem 1, we can retrieve at most 2.86 node-independent routes. One way to remedy the low degree issue is to use super nodes for source nodes and destination nodes. For instance, in Figure 2(b), $S1$ and $S2$ are grouped into a super source node S , and $T1$ and $T2$ are grouped into a super destination node T . Consequently, one node-independent shortest path algorithm for S and T can compute four node-independent shortest routes as increasing node degree of S and T .

The spatial network organization of a place has an extremely important effect on the way people move through space and time. Evacuation route planning involving large numbers of evacuees has a well defined evacuation zone (EZ) (e.g., entire cities or coastal plains), making it possible to find the spatial movement patterns on transportation networks. For example, in Figure 3(a), the circular EZ encloses 12 nodes and all the travelers on these nodes need to move out of the circle. The network has one unit of capacity with one unit of travel time and two evacuees per node. Figure 3(b) shows the network model for the EZ; The nodes inside of the EZ are $(A, B, C, D, E, F, G, H, I, J, K, L)$ and the nodes outside of the EZ are $(X1, X2, X3, X4, X5, X6, X7, X8, X9, X10, X11, X12)$. The nodes in the EZ are divided into two groups by a DBN-cut. The outside nodes (A, B, C, F, G, J, K, L) have more shorter available routes compared to the inside nodes (D, E, H, I) , reflecting an “outer first, inner last” flow pattern. That means, after evacuees from boundary areas move out of outer boundary areas, there is a secondary wave of evacuees from inner regions into boundary areas that will prepare to move out of the EZ. To characterize this pattern, dartboard network structure is defined as a network organization partitioned according to Dartboard Network Cuts (DBN-cuts), shown in Figure 3(b). To explain how this structure happens, look at the arrows in Figure 3(b). Given the EZ and number of evacuees, the evacuation plan needs to maximize the number of evacuees using the available shortest routes shown as arrows at each time step $t \in T$. As can be seen in Figure 3(b), the outer group (A, B, C, F, G, J, K, L) is poised to flee first from the EZ to maximize the number of evacuees, followed by the inner group (D, E, H, I) , which takes its place and is similarly set to flee to maximize the number of evacuees again. We define a dartboard network cut (DBN-cut) as a cut in the flow of travelers in an evacuation network such that all the travelers in a single group are removed at the same time.

Theorem 2. *In a dart board network structure with FIFO property and a limited capacity constraint, a maximal dynamic flow algorithm for evacuation planning moves the outside nodes first and goes to inside nodes incrementally.*

Proof. Assume that evacuees from inside nodes arrive at the destination before evacuees at outside nodes. This means that some outside nodes must have had to wait to exit the boundary area in order to make sure there is available capacity for evacuation of inside nodes. Otherwise, there would not have been enough capacity available for the inside nodes to exit the EZ through the outer boundary

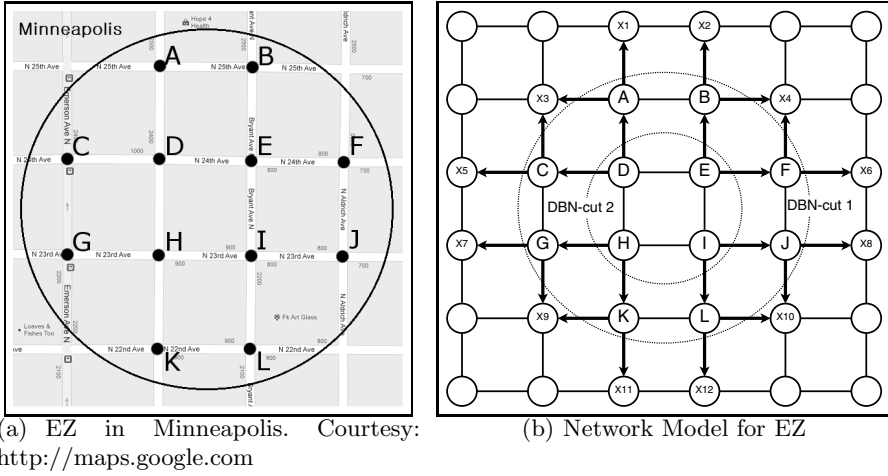


Fig. 3. Dartboard network structure in road networks

Table 1. Evacuation route plan based on dartboard network structure in Figure 3(b)

Group	Source	# of Evacuees	Route Node Id(Time)	Arrival Time	Group	Source	# of Evacuee	Route Node Id(Time)	Arrival Time	
1	A	1	A(0)-X1(1)	1	1	K	1	K(0)-X9(1)	1	
		1	A(0)-X3(1)				1	K(0)-X10(1)		
	B	1	B(0)-X2(1)			1	L	1		L(0)-X10(1)
		1	B(0)-X4(1)					1		L(0)-X12(1)
	C	1	C(0)-X3(1)		2	D	1	D(0)-A(1)-X1(2)		
		1	C(0)-X5(1)				1	D(0)-C(1)-X5(2)		
	F	1	F(0)-X4(1)		1	E	1	E(0)-B(1)-X2(2)		
		1	F(0)-X6(1)				1	E(0)-F(1)-X6(2)		
G	1	G(0)-X7(1)	1	H	1	H(0)-G(1)-X7(2)				
	1	G(0)-X9(1)			1	H(0)-K(1)-X11(2)				
J	1	J(0)-X8(1)	1	I	1	I(0)-J(1)-X8(2)				
	1	J(0)-X10(1)			1	I(0)-L(1)-N12(2)				

area. This implies an increase in the total time required to evacuate all people, thereby violating our objective to minimize the evacuation time.

Table 1 shows the results of an evacuation route plan with dart-board network structure. Each row in the table describes the schedule of a group of evacuees moving together to arrive at destinations at time step $t \in T$. During each iteration, the algorithm tries to group the node-independent routes and maximize the number of evacuees. For example, group 1 aggregates 16 node-independent shortest routes for 32 evacuees and group 2 aggregates 8 node-independent shortest routes for 16 evacuees. Note that each group aggregates evacuation routes on different spatial locations and reaches destinations at the same time.

Given the number of routes and the evacuation time, our principal objective is to maximize the number of evacuees for each time step $t \in T$. To achieve this end, an evacuation routing algorithm attempts to maximize the available shortest routes at each time step $t \in T$. Given this basic assumption, in each time step t , many node-independent routes may exist to maximize the number of

available shortest routes. In the next subsection, we introduce our new algorithms to efficiently create DBN-cuts on evacuation networks.

3.2 DBNC-ERP Algorithm

In this subsection, we describe our node-independent shortest paths approach for a dartboard network structure. A naive approach is to enumerate all available routes and remove node-dependent routes. However, the search space becomes exponential for combinations of the available multiple routes. General approaches for node-independent shortest paths construct a shortest path tree (SPT) and check the node dependency for each route [20,19,12]. The SPT of order n nodes has size $n - 1$, resulting in reduced search space by examining the boundary nodes [13]. In our problem, there are many source nodes to traverse in order to reach destination nodes. Instead of a SPT, we consider a shortest *forest* where each tree has a different root to handle multiple source nodes and iteratively choose node-independent shortest routes having available capacity. A forest of order n with k roots (or source nodes) has size $n - k$ since not every forest shares nodes. We allow SPTs in the forest to share the same destinations because evacuees from different sources may reach the same destination. For each route, possible waiting time at each node is considered due to limited capacity constraint. Algorithm 1 shows a way to identify the evacuation routes with node-independent shortest paths. The input for the pseudo code is an evacuation transportation network consisting of nodes, edges, source nodes, sink nodes,

Algorithm 1. Pseudo code for DBNC-ERP

Inputs: - A set of nodes N and edges E with capacity constraints C

- Each edge $e \in E$ has a travel times t .

- A set of source nodes S including initial evacuee occupancy O and a set of destination nodes D

Outputs: Evacuation plan including route schedules of evacuees on each route r

DBNC-ERP Algorithm:

- 1: **while** any source node $s \in S$ has evacuees **do**
 - 2: group all source nodes with a super source node ss and group all sink nodes with a super sink node sd .
 - 3: construct a shortest forest from S to D based on ss and sd . Every tree can share the destination D .
 - 4: find all routes R that are shortest node-independent paths from S to D .
 - 5: **for** $r \in R$ **do**
 - 6: compute the minimum route capacity C_{min} with the edge and node capacity c along the route r .
 - 7: evacuee flow $f = \min(\text{number of remaining evacuee at } s \text{ in } r, C_{min})$.
 - 8: reduce the node and edge capacity c along the route r using f .
 - 9: remove evacuees from O using f .
 - 10: **end for**
 - 11: **end while**
-

and capacity constraints. The output is an evacuation route schedule containing a sequence of nodes and edges. All source nodes and destination nodes are grouped by two super nodes to increase the node degree (Line 2). A shortest forest is constructed to find node-independent routes (Line 3,4). After retrieving the node-independent routes, the capacity constraints are applied to these routes and available routes for evacuees are chosen (Line 5,6). The next step is to reduce node and edge capacities along the routes (Line 7,8) and repeat the above process until we finish finding the routes for all remaining evacuees.

The DBNC-ERP algorithm based on node-independent routes may need several iterations before obtaining available routes at each time step $t \in T$. Our greedy approach is related to an aspect of DBN-cuts that attempts to maximize the evacuation routes for each time step t . Line 4 in Algorithm 1 evaluates all available routes based on the “share-nothing” property of node-independent routes and reduces iterations for constructing the shortest forest.

4 Algebraic Cost Model of DBNC-ERP

The goal of this section is to present cost models for DBNC-ERP for estimating computational cost based on node-independent routes. The transportation road network can be modeled as a grid-like network that has many alternative shortest routes [23,18]. In general, there are at least two node-independent routes between any pair of nodes [14]. In our analysis, we use 2 as a lower bound of road network connectivity. Assume that n is the number of nodes, m is the number of edges, and p is the number of evacuees. The DBNC-ERP iteratively chooses k node-independent shortest routes and reserves the capacity for these routes. In the worst case, one evacuee can traverse the route, resulting in p/k iterations. DBNC-ERP constructs a shortest forest using a modified Dijkstra’s algorithm and super nodes. The worst case computational time for a dense graph is $O(n^2)$. For sparse networks, Dijkstra’s algorithm can be implemented in time $O(n \log n)$ [4,17]. Basically, the node-independent shortest routes are computed as the same bounds for Dijkstra’s algorithm [21,6]. In our approach, we put super nodes to group the source nodes and destination nodes, then construct a shortest forest instead of a SPT. Capacity constraint checking and updating takes $O(kn)$ for k node-independent shortest routes. The cost model of the DBNC-ERP algorithm is $O((p/k) \cdot n \log n)$. In transportation road networks, we can compute the lower bound of DBNC-ERP as $O((p/2) \cdot n \log n)$.

The cost model shown above is the strictly lower bound. This is because this model does not consider the DBN-cuts which group source nodes in different spatial locations, leading to increase degree of a source node. If the network has sufficiently large numbers of destination nodes, then the number of node-independent routes is bounded by the degree of a super source node, according to Theorem 1. This point is easily illustrated by a circular evacuation zone in Figure 3. From the boundary of the EZ to its center, DBNC-ERP incrementally groups source nodes using DBN-cuts and attempts to find the available evacuation routes for each source of the group. This reduces iterations of DBNC-ERP

by the number of DBN-cut groups. However, the number of groups for DBN-cuts is highly dependent upon the underlying network structure of the EZ.

5 Experimental Evaluation

Figure 4(b) shows our experimental setup. Our experiments used a Minneapolis, MN road map consisting of 8,868 nodes and 24,126 edges, taken from TIGER/Line [2]. The software was implemented in Java 1.7 with 1 GB memory run-time environment. All experiments were performed on an Intel Core i7-2670QM CPU machine running MS Windows 7 with 8 GB of RAM. We used two evacuation zones (EZs): one for a circular area and the other for a riverside area. Given the location of an incident and its scope R , we defined a circular EZ as the circular area centered at the incident with radius R and a riverside EZ (or coastal EZ) as the buffer area with R adjacent to rivers. We tested three different approaches. The first two are CCRP [15] and DBNC-ERP with node-independent shortest paths (DBNC-ERP with NI). The third approach was DBNC-ERP with a shortest forest, as a candidate for DBNC-ERP to attempt to maximize the number of evacuees at each time step $t \in T$. The property of node-independent routes is easily exploited to reduce the computational time by reducing each iteration. However, the DBNC-ERP algorithm with node-independent shortest paths uses two route scans to evaluate the availability: one scan for node dependency and the other for capacity constraints. Intuitively, the forest for node-independent routes displays partially node-independent. We may relax the node-independent constraints and remove the node-dependency check for the forest. We call this method DBNC-ERP with a shortest forest. The strength of this approach is that it reduces route checking time when the number of available routes is large. The main disadvantage is that it may yield false-positive node-independent routes, which will be removed during capacity checking time.

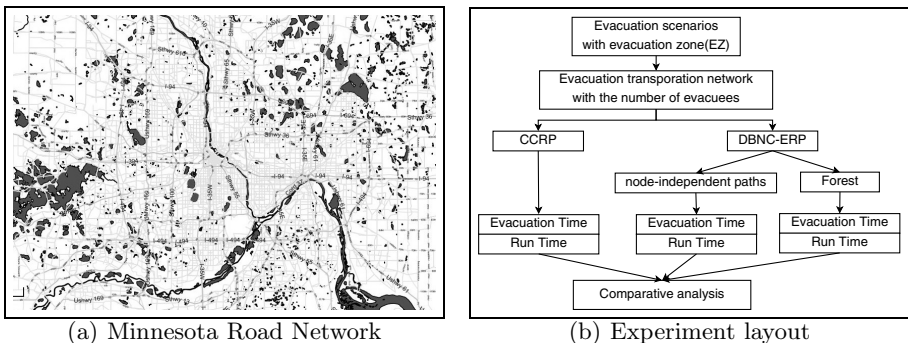


Fig. 4. Experiment setup for evacuation routing planning

5.1 Experimental Observation and Results

Experiment 1: Effect of the number of evacuees

The purpose of the first experiment was to evaluate the effect of number of evacuees on the performance of the algorithms. We fixed the number of source and destination nodes and multiplied the evacuees of each node. The experiment was done using networks of 246 source nodes, 109 destination nodes, and 1,847 nodes for a circular EZ. We incrementally increased the number of evacuees from 766,123 to 3,064,492. Figure 5(a) shows that the two DBNC-ERP approaches outperform CCRP. As increase of number of evacuees, the performance gap also increases. This is because the DBNC-ERP approaches group the node-independent routes to minimize the iterations. DBNC-ERP with a forest shows slightly better performance compared to DBNC-ERP with IN due to the longer node-dependency checking time. Figure 5(b) shows that all three algorithms were not distinguished in terms of evacuation time results. As the number of evacuees grows, the egress time increases.

Experiment 2: Effect of the number of source nodes

The second experiment evaluated the effect of the number of source nodes on the performance of the algorithms. We fixed the number of destination nodes and the number of evacuees. To increase the number of source nodes, source nodes were made to share the evacuees to new source nodes. The experiment was done using networks of 109 destination nodes, 1,847 nodes for a circular EZ, and 766,123 evacuees. We incrementally increased the number of source nodes from 246 to 984. Figure 6(a) shows that number of source nodes has little effect on algorithm performance. Nevertheless, the two DBNC-ERP approaches run faster than CCRP.

Experiment 3: Effect of the number of destination nodes

The third experiment evaluated the effect of the number of destination nodes on the performance of the algorithms. We fixed the number of source nodes,

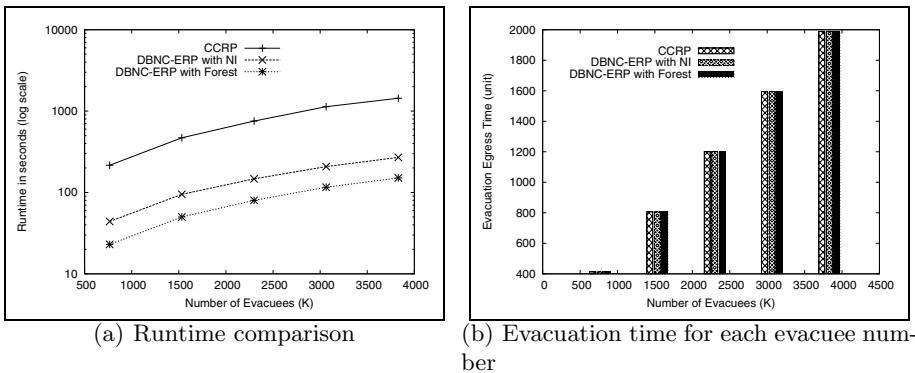


Fig. 5. Effect of the number of evacuees

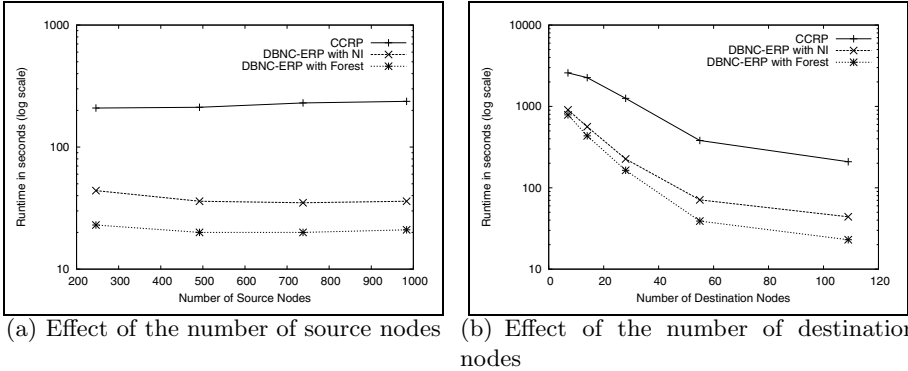


Fig. 6. Effect of the number of source and destination nodes

and the number of evacuees and decreased the number of destination nodes. The experiment was done using networks of 246 source nodes, 1,847 nodes for circular EZ, and 766,123 evacuees. Figure 6(b) shows that as the number of destination nodes grows, the runtime decreases. As the number of destination nodes increases, the performance gap also increases according to Theorem 1.

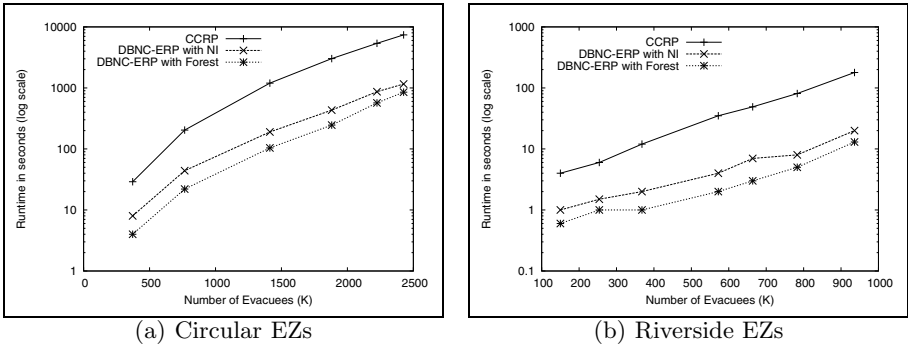


Fig. 7. Scalability on different EZ shapes

Experiment 4: Scalability for large network datasets

The fourth experiment evaluated scalability for large network datasets. We incrementally increased the radius of the circular EZ from 5km to 30km. Figure 7(a) shows that the runtime of DBNC-ERP scaled well to these large network sizes. These results show that runtime can be reduced by up to 80%

Experiment 5: Effect of shape of EZ

The fifth experiment evaluated the effect of the shape of the EZ on the performance of the algorithms. If we put the destination nodes as boundary nodes of the EZ, a circular EZ will have a least possible number of boundary nodes

due to its small surface area. If the EZ shows irregular shape (e.g., coastal areas), the number of boundary nodes increases as the surface area of the EZ increases. In our experiment, we chose evacuation zones along rivers as shown in Figure 1(a) and incrementally increased the length of the non-circular EZ to cover the entire length of MN rivers. Results showed that the two DBNC-ERP approaches ran faster on the riverside EZ (Figure 7(b)) than on the circular EZ (Figure 7(a)). This is because the number of boundary nodes of the irregular shaped EZ is greater than of the circular EZ. Our results show that the runtime can be reduced by up to 90% when our approach is applied to irregularly shaped evacuation zones.

Experiment 6: Effect of spatial network structure of EZs

The last experiment explored the effect of different spatial structures of EZs. We chose five different coastal or isolated regions taken from OpenStreetMap [3] and assigned a synthetic population for each EZ. Table 2 shows that the maximum speed up in DBNC-ERP algorithms is bounded by the number of dartboard network cuts enclosing the EZ (i.e., the number of destination nodes). Once again, the fewer the destination nodes (e.g., Key West and Galveston) for EZs, the fewer the node independent routes to speed up.

Table 2. Experimental Result for other regions

Region	Runtime reduction	# of nodes	# of edges	# of dart board network cuts	# of evacuees
Key West,	64%	1,291	3,809	3	25,820
Galveston, TX	82%	4,146	12,368	3	36,675
Jackson, WY	85%	673	1,696	13	9,422
Cape Cod, MA	92%	32,257	80,438	30	225,799
San Francisco, CA	96%	16,409	48,058	78	810,084

5.2 Discussion

The proposed DBNC-ERP algorithm advances the state of the art computational techniques for evacuation route planning. The proposed algorithm achieves a significant computational performance gain over current techniques. This improvement was obtained using three key features of the underlying road networks: (1) FIFO with limited capacity, (2) “outer-first” flow pattern, and (3) dartboard network structure.

The first, two features implicitly assume that the risk in a given EZ is distributed uniformly. The DBNC-ERP algorithm uses an incremental strategy for such EZs, where people in the outer region of the EZ are evacuated first. This leads to reductions in overall evacuation time. However, an incremental strategy may not be suitable for EZ’s with non-uniform risk (e.g. point based threats such as bombs). Typically, in such scenarios, the evacuation proceeds in phases, where evacuees on the inner ring of nodes are evacuated first. This type of protocol may violate the “outer-first” and FIFO assumption. We plan to explore such multi-phase evacuation scenarios in the future.

The third feature, dartboard network structure, allows the DBNC-ERP algorithm to reserve capacities along k paths per iteration. Here, k is the number of node-independent routes. During the execution of the algorithm, parameter k manifests itself as the number of dartboard network cuts in the given EZ. Our results to date provide evidence for a correlation between the size of dartboard network cuts and performance gain. For example, the proposed algorithm achieved a reduction of 92% in runtime for Cape Cod which had a cut set of size 30. On the other hand, the Key West dataset with a cutset size of 3, allowed a reduction of 64%.

Due to time limitations, we have only used five geographic areas for preliminary evaluation of the proposed algorithm. In the future, we plan to test our algorithm on a larger number of geographic areas to characterize this correlation between the spatial structure (dartboard network cuts) of the road network and the computational running time.

6 Conclusion and Future Work

Evacuation route planning for large transportation networks is becoming increasingly important for dealing with man-made and natural disasters, such as hurricanes, terrorist acts, and nuclear accidents. An important component of evacuation planning methods is the ability to account for capacity constraints of the road network with manageable computational cost. In this paper, we introduced dartboard network structure to reflect evacuee flow pattern for common evacuation scenarios by exploiting the spatial structure of the road network. Based on dartboard network structure, our DBNC-ERP algorithm partitions the network using dartboard network cuts (DBN-cuts) and groups source nodes in different spatial locations to maximize the number of evacuees. We also showed the cost model to explain how to reduce the computational cost based on dartboard network structure. Experimental evaluation of DBNC-ERP demonstrated significant improvements over previous work.

In the future we plan to further explore the observed relationship between the spatial structure of the road network and computational performance gain. Also, we plan to study computational techniques for evacuation planning scenarios with non-uniform risk. Additionally, we would like to explore evacuation route planning algorithms for cloud environments which can handle much larger networks.

Acknowledgments. We would like to thank the National Science Foundation and the US Department of Defense for their support with the following grants: NSF grant (grant number NSF III-CXT IIS-0713214) and USDOD grant (grant number HM1582-08-1-0017). We are particularly thankful to Giscience reviewers for their helpful comments. We also extend thanks to the University of Minnesota Spatial Databases and Spatial Data Mining Research Group for their comments. We would like to thank Kim Koffolt for improving the readability of this paper.

References

1. The New York Times, HURRICANE ANDREW: When a Monster Is on the Way, It's Time to Get Out of Town. In: Texas, a Line of Cars 50 Miles Long (August 26, 1992), <http://goo.gl/hq0EH> (retrieved April 2012)
2. U.S.Census Bureau - TIGER/Lines, <http://goo.gl/P6Ye7> (retrieved January 2012)
3. OpenStreetMap, <http://goo.gl/Hso0> (retrieved April 2012)
4. Ahuja, R., Magnanti, T., Orlin, J., Weihe, K.: Network flows: theory, algorithms and applications. Prentice Hall (1993)
5. Ben-Akiva, M., et al.: Development of a deployable real-time dynamic traffic assignment system: Dynamit and dynamit-p users guide. Intelligent Transportation Systems Program. Massachusetts Institute of Technology (2002)
6. Bhandari, R.: Survivable Networks: Algorithms for Diverse Routing. Kluwer Academic Publishers, Norwell (1998)
7. Fleischer, L., Skutella, M.: Quickest flows over time. *SIAM Journal on Computing* 36, 1600–1630 (2007)
8. Ford, D., Fulkerson, D.: Flows in networks. Princeton university press (2010)
9. Gastner, M., Newman, M.: The spatial structure of networks. *The European Physical Journal B-Condensed Matter and Complex Systems* 49, 247–252 (2006)
10. Hamacher, H., Tjandra, S.: Mathematical modelling of evacuation problems: State of the art. In: *Pedestrian and Evacuation Dynamics*, pp. 227–266. Springer (2002)
11. Hillier, F., Lieberman, G., Hillier, M.: Introduction to operations research. McGraw-Hill (1990)
12. Kleinberg, J.M.: Approximation algorithms for disjoint paths problems. Ph.D. Dissertation, Dept. of CS., Massachusetts Institute of Technology (1996)
13. Korf, R., Zhang, W., Thayer, I., Hohwald, H.: Frontier search. *Journal of the ACM (JACM)* 52, 715–748 (2005)
14. Levinson, D., Yerra, B.: Self-organization of surface transportation networks. *Transportation Science* 40, 179–188 (2006)
15. Lu, Q., George, B., Shekhar, S.: Capacity Constrained Routing Algorithms for Evacuation Planning: A Summary of Results. In: Medeiros, C.B., Egenhofer, M., Bertino, E. (eds.) *SSTD 2005*. LNCS, vol. 3633, pp. 291–307. Springer, Heidelberg (2005)
16. Mahmassani, H., Sbayti, H., Zhou, X.: Dynasmart-p: Intelligent transportation network planning tool: Version 1.0 users guide. Maryland Transportation Initiative, University of Maryland, College Park, MD (2004)
17. Schrijver, A.: Combinatorial optimization. Springer (2003)
18. Shekhar, S., Chawla, S.: Spatial databases: a tour. Prentice Hall, Upper Saddle River (2003), 7458
19. Sidhu, D., Nair, R., Abdallah, S.: Finding disjoint paths in networks. *ACM SIGCOMM Computer Communication Review* 21, 43–51 (1991)
20. Suurballe, J.: Disjoint paths in a network. *Networks* 4, 125–145 (1974)
21. Suurballe, J., Tarjan, R.: A quick method for finding shortest pairs of disjoint paths. *Networks* 14, 325–336 (1984)
22. Wardrop, J.: Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers* 2(1) (1952)
23. Xie, F., Levinson, D.: Measuring the structure of road networks. *Geographical Analysis* 39, 336–356 (2007)
24. Zhou, X., George, B., Kim, S., Wolff, J., Lu, Q., Shekhar, S., Nashua, O., Team, G.: Evacuation planning: A spatial network database approach. *Bulletin of the Technical Committee on Data Engineering* 33(2), 26 (2010)

Hybrid Geo-spatial Query Methods on the Semantic Web with a Spatially-Enhanced Index of DBpedia

Eman M.G. Younis, Christopher B. Jones, Vlad Tanasescu, and Alia I. Abdelmoty

School of Computer Science & Informatics, Cardiff University,
CF24 3AA, United Kingdom

{E.Younis, C.B.Jones, V.Tanasescu, A.I.Abdelmoty}@cs.cardiff.ac.uk

Abstract. Semantic Web resources such as DBpedia provide a rich source of structured knowledge about geographical features such as towns, rivers and historical buildings. Retrieval from these resources of all content that is relevant to a particular spatial query of, for example, containment or proximity is not always straightforward because there is considerable inconsistency in the way in which geographical features are referenced to location. In DBpedia some geographical feature instances have point coordinates, some have qualitative properties that provide explicit or implicit locational information via place names, and some have neither of these. Here we show how structured geo-spatial query, a form of question answering, on DBpedia can be performed with a hybrid strategy that exploits both quantitative and qualitative spatial properties in combination with a high quality reference geo-dataset that can help to support a full range of geo-spatial query operators.

Keywords: GIS, maps, question answering, query, linked data, sparql.

1 Introduction

The Web can be regarded as a rich source of geographical information but much of that information can be difficult to retrieve because it is embedded in natural language text. Conventional search engines can access documents that contain place names in a user's query but unless the required content is a yellow pages listing, which may be picked up with so-called local search methods, it remains for the user to sift through the retrieved documents to find relevant information. Research into spatially-aware search engines has produced systems that can improve the quality of retrieval results but those results are still typically unstructured text documents (Purves et al, 2007). Semantic Web technologies have been motivated by the objective of machine-readable access to structured data on the Web, which holds the promise of much more focused responses to user queries. The SPARQL query language for example can be used to formulate queries on RDF (Resource Description Framework) data records on the Web in the form of subject-predicate-object triples. If semantic equivalence between the data items in different triple stores can be asserted then they can be linked together to support more complex queries across multiple RDF data sources. As increasing amounts of geo-referenced information become encoded in this way the Semantic Web is becoming a valuable source of structured geo-information.

Probably the currently richest source of geo-referenced information on the Web, with regard to the semantics, is to be found in Wikipedia and its Semantic Web version DBpedia (an RDF resource: <http://dbpedia.org/>) which contains hundreds of thousands of entities that are geo-referenced with point geometry, but not with lines or polygons. Increasing quantities of digital map data that can complement the geo-semantic content of RDF resources such as DBpedia are also appearing, with OpenStreetMap (OSM) being perhaps the most substantial source of freely available topographic features encoded as points, lines and areas. OSM has been converted to RDF format and links have been determined between some of its geometric features and corresponding topics in DBpedia in the LinkedGeoData project (Stadler et al. 2011). That project has exploited the links with a mapping system that supports pan and zoom of maps annotated with DBpedia features, but not structured spatial query.

The work presented in this paper is motivated by the objective of supporting high quality spatial query to rich semantic RDF content, such as DBpedia, which has many records describing the semantics of geographical features but has limited geo-spatial data. Thus we wish the user to be able to perform typical geo-information queries such as to find specified types of content within named regions, such as a city, or within a specified distance of the centre or the boundary of a region and to find features that hold overlap and crossing relations with a reference place or feature. Such queries cannot be performed using only the single point geometry associated with geo-referenced features in DBpedia. To achieve our objective we maintain a high quality, detailed reference digital map dataset for the entire region of interest, so that queries that name any feature in the region can be instantiated with the relevant feature geometry. The reference geo-data are stored in a spatially indexed database in combination with a spatial index of geo-referenced entities in RDF content, i.e. DBpedia.

The approach may be seen as a spatially intelligent index of Semantic Web content comparable with the role of an inverted index in a conventional web search engine, but in the work presented here we only currently access the single RDF resource of DBpedia. The analogy is with regard to the need for fast access to Web resources that satisfy the user's query constraints, which in our case can include spatial qualifiers. Unlike a conventional search engine for web documents which returns the URLs (uniform resource locators) of matching documents, we use the URIs (uniform resource identifiers) retrieved from the local index to formulate SPARQL queries on the DBpedia endpoint (a URL) if that is required to provide an explicit answer to the query.

In the present version of our experimental system the spatial index of DBpedia content is based on the point coordinates associated with DBpedia instances. Because not all geographical DBpedia instances actually have coordinates, we exploit other qualitative properties of the instances to associate them with contained places. Thus there are multiple properties that name the containing town, city or other administrative area, such as *dbpo:administrativeDistrict*¹, *dbpo:location* and

¹ The prefix *dbpo* stands for <http://dbpedia.org/ontology/>, while *dbpedia* stands for <http://dbpedia.org/resource/> and *dbpp* for <http://dbpedia.org/property/>

dbpo:locatedInArea in which the property name is either implicitly or explicitly spatial. These types of properties are particularly valuable in processing containment queries where the user-specified containing place may match the object of one of these DBpedia properties. There are also properties that indicate other spatial relations such of cardinal direction and proximity, but these latter properties have no consistent spatial interpretation and are not used in any consistent way. It is also the case that, just as many DBpedia place instances do not have coordinates, there are many place instances that do not have properties that imply containment at a useful granularity. For purposes of containment search our approach is therefore a hybrid one that combines exploitation of the qualitative spatial properties with results obtained from geo-spatial processing methods that test for containment of DBpedia point coordinates within polygons provided by the reference geo-data. Exploitation of coordinates and spatial properties cannot be guaranteed to find all place instances that may be relevant to a containment query, as some instances have neither of these types of property. Successful access to such instances may depend upon automated geo-referencing using methods such as those described in De Rouck et al. (2011), which was applied to Wikipedia articles. Our system is designed on the assumption that this will be achieved in the future, enabling DBpedia instances to be maintained in the local spatial index.

It should be noted that because the approach we present depends for its effectiveness upon the presence of rich and detailed digital map data corresponding to the RDF semantic content, we employ a national mapping agency dataset, with consistent high quality spatial coverage, to demonstrate the approach for a single country. As OSM evolves it may well be able to serve that role and is of course international in coverage.

In the remainder of the paper we summarise related work in section 2 before providing an overview in section 3 of the architecture of our experimental system. Section 4 presents some experimental results that demonstrate the effectiveness of the approach including an analysis of the availability of coordinates and qualitative spatial containment properties. The paper concludes in Section 5 with a summary of the progress to date and directions for further development of public access spatial query of geographical information on the Semantic Web.

2 Related Work

Our work on structured query of geographic information can be regarded as a form of geographic question answering system (GQAS), but it differs from much existing work in that area in focusing on structured data rather than free text and in the use of geo-spatial processing in addition to exploiting some qualitative data. We review first briefly some work in this area that is largely based on language processing. The START system (Katz and Lin, 2002; Lin and Katz, 2003) accepts natural language questions and can provide some properties of geographic places such as their population and distances between places, but it is not able to satisfy typical geo-spatial questions regarding proximity and topological relationships between places.

The Geo-Logica system (Waldinger et al, 2004) incorporates an automated deduction system with spatial and temporal reasoning capabilities, whereby having formulated the natural language query in a logical form, geographic information is extracted from text documents from various sources. It cannot process spatial relations explicitly or compute spatial properties from geo-data. A voice activated GQAS is proposed in Luque et al (2008) that allows speech input of questions about Spanish geography. It is based on language analysis of free text Web resources in combination with place name gazetteers and a training corpus of geographical questions. The QUASAR system (Buscaldi, 2007) uses language processing to access free text sources, including use of Wikipedia, to extract geographic information, with a focus on word sense disambiguation. The GeoCLEF and GikiCLEF events have resulted in publication of geographic question answering systems but these are mostly based on information extraction from free text documents. The work described by Hartrumpf and Leveling (2010) combines text information extraction with geographical information retrieval (GIR) methods that work with a spatial index of documents, and does exploit DBpedia, alongside Wikipedia, as a source, but it converts the RDF to natural language expressions for processing by the non-GIR methods. Mishra et al (2010) employ the user's query to retrieve documents from a search engine that are then subject to information extraction, results of which can be viewed on a map.

The increasing quantity of Semantic Web resources, including the Geonames gazetteer, OSM and DBpedia, has led to several initiatives to provide spatially enabled access to their content. The LinkedGeoData project (Stadler et al, 2011) contributed to the transformation of OSM to RDF and has presented methods to determine links between map features in OSM and equivalent instances documented in DBpedia, as well as between OSM and Geonames. Their matching is based on a combination of the Jaro-Winkler string distance between the text of the respective place names and the geographic distance between the entities. They have illustrated exploitation of the links with an interactive map that supports pan and zoom but that application does not support geo-spatial query with conventional spatial relationships. Linking between equivalent entities is a critical issue in exploiting Semantic Web data. Examples of other work on linking geo-data on the Semantic Web are Hahmann and Burghardt (2010), which uses Levenshtein string distance, and Sala and Harth (2011) which employs the Hausdorff distance to establish similarity between spatially extensive linear or polygonal features. We employ similar methods in our work to establish links between DBpedia place instances and geo-features in our reference geo-data store.

The availability of geospatial data on the web has motivated various developments to enhance SPARQL, the main language for access to RDF, with spatial functionality which may be supported by the various triple stores of RDF content. GeoSPARQL reflects a W3C supported initiative to create such a language (Battle and Kolas, 2011). It provides a full range of spatial operators accessed via a spatial index of the corresponding RDF data store. Rather than modifying the SPARQL language for geo-spatial query, Brodt et al (2010) present an approach that confines spatial functionality to SPARQL filter functions. A hybrid approach is presented in Della Valle et al (2010) in which spatially enhanced SPARQL queries are mapped to a

PostGIS spatial database that implements spatial indexing and spatial query operators. They demonstrated the approach with queries that employ a mix of polygonal data from OSM data with point referenced data such as from DBpedia. Our work differs from such approaches in creating a centralized spatially-enabled index of Semantic Web content, in the manner of a web search engine, and employing a locally-stored reference geo-dataset to mediate queries that require line and polygonal representations of named features. We implement hybrid query methods that exploit qualitative spatial properties in addition to quantitative geo-data. The system also performs queries on an RDF triple store if required

Because many Semantic Web resources such as Geonames, OSM and DBpedia are volunteered resources, contributed to by individuals with only informal procedures for validating the content, there can be considerable variation in the quality and coverage of the data (see for example Hackley 2010 and Mooney et al 2010, for some analyses of OSM). Although OSM is improving greatly in coverage, for the purposes of our experimental system we have used a national mapping agency dataset which, while restricted to a single country for purposes of our experimental system, has high quality linear and polygonal geometric representation of many map features including city boundaries. For international coverage, OSM clearly has tremendous potential to serve a similar function.

3 System Architecture

The key components of the architecture we present (see Figure 1) consist of a query interface, a query processor, a local spatially indexed geo-data repository (referred to as the reference geo-data or reference geometry), a local spatially indexed store of place URIs extracted from DBpedia, and access via SPARQL to the external RDF store which is DBpedia.

The current version of the experimental user interface supports structured query for question answering enabling the user to enquire about properties of places, and to find places subject to spatial constraints of containment, crossing and proximity. When enquiring about properties of places the user is given a drop down menu that lists the DBpedia properties associated with the place the user names. A SPARQL query to DBpedia is used to retrieve the answer. For spatial queries, that may include a feature type constraint, the following methods have been implemented for comparative purposes:

- Within distance of a point, line or area geometry object, where the buffer is created from reference geo-data;
- Crossing a line or area feature, where the DBpedia instance may be represented by reference geo-data, such as a line, and the feature is also represented by reference geo-data;
- Containment using a hybrid approach that combines the results from geo-data search in which the containing area is represented by reference geometry, with results from a SPARQL query that uses qualitative spatial (or implied spatial) properties.

In all the above, feature types may be selected from available types used with DBpedia.

The query processor performs tasks of query planning according to the nature of the user query, generation of a query footprint, retrieval of relevant URIs from the local spatial index of DBpedia content and formulation and execution of SPARQL queries to the external RDF store, followed by return of the result to the user interface.

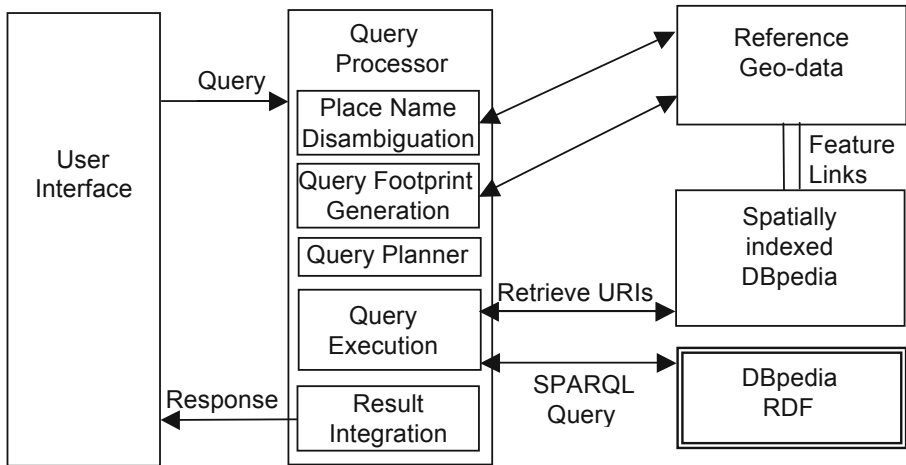


Fig. 1. Architecture of system for geo-spatial structured query of DBpedia

Query planning classifies a user query into one of three types:

1. A non-spatial query selecting by properties of places: requires SPARQL access to DBpedia only;
2. A proximity (within distance) and crossing/overlap query that selects relative to named geo-features: requires reference geo-data, and spatial access to local DBpedia index and may require a SPARQL query for DBpedia properties;
3. A containment query that selects within a named region: combines results from a spatial query that uses reference geo-data, with results of SPARQL queries using spatial containment properties. May require SPARQL query to access specialised properties of retrieved place instances.

The first situation is relatively trivial in that it requires only a SPARQL query to retrieve required properties of the named DBpedia instances. The second situation for proximity and overlap/crossing queries requires the creation of a query footprint based on the local reference geo-data store. Thus, for example, a query for DBpedia instances within a specified distance of a named river requires access to the river geometry from the local geo-data. This is then used to issue a query to the local

spatial index of DBpedia instances corresponding to the river objects, in combination with constraints on feature type (which is recorded in the index). If some specialised property of the retrieved places is required then a SPARQL query to DBpedia is performed using the URIs of relevant instances that were previously retrieved.

The third situation of a containment query relative to a named regional place (such as a city) will find results using both local spatial indexing of geo-referenced content and deduction of containment from appropriate DBpedia properties via a SPARQL query. The results are then merged, as there may be some duplication for instances that maintain both coordinates and the implied spatial properties. The use of the local geo-spatial data is similar to the previous strategy in that, having disambiguated the query place name, a boundary for the place is retrieved from the local geo-data if there is one. Our local geo-data is notable for maintaining boundaries for a large number of named settlements. The boundary is then used to perform a PostGIS query on the spatially indexed DBpedia content in combination with feature type constraints.

Identification of properties that specify or imply containment is a semi-automated process. For given feature classes that correspond to regions of space, such as a city, we select representative, well known place instances in DBpedia and retrieve all properties for which the respective place is the object of the property. The resulting properties are then filtered manually to remove those which do not in fact imply spatial containment, such as *dbpedia:birthPlace*. To perform a containment query the resulting list can be used to filter the results from a SPARQL query in which the object is the named place and the subject is constrained to a user specified type. Our strategy is to perform the filtering on the results of the SPARQL query, in the query processor, though it would also be possible to formulate a more complex SPARQL that included this filtering process.

The results from both types of containment query are then merged and if the query requires some other named property of the retrieved places then a further SPARQL query is executed to retrieve those properties of the previously found place instances.

The local store of reference geo-data consists in our experimental system of UK Ordnance Survey named features which are stored in a PostGIS spatial database that provides OGC compliant spatial queries for topological relations as well as distance (buffer) searches. When the query processor obtains a reference toponym from the user's query this must be matched to a name in the local reference geometry database (which may require disambiguation via the user interface). Using PostGIS we have a full set of OGC spatial operators and by maintaining the index locally we have faster response than if the reference geo-data needed to be retrieved from a spatially-enabled SPARQL endpoint.

The local store of georeferenced DBpedia content was obtained by performing SPARQL queries on DBpedia to access all places of particular types, which were filtered via their coordinates, where present, to confine much of the content for our experimental system to the British Isles. To do this requires knowledge of all feature classes of geographical instances. Our approach to this was to identify representative instances of different broad classes of place and to ascend the hierarchies of their

respective class properties in order to identify relevant parent classes. For each general category we then performed SPARQL queries to retrieve all instances of these parent classes and their children as illustrated by the following query for the category Museum.

```
define input:inference
"http://dbpedia.org/resource/inference/rules/yago#"
PREFIX yago:      <http://dbpedia.org/class/yago/>
PREFIX dbpo:     <http://dbpedia.org/ontology/>
SELECT DISTINCT ?s ?lat ?lon ?geom ?point
FROM <http://dbpedia.org>
WHERE {
  {?s a dbpo:Museum }
  UNION
  {?s a ?t . ?t rdfs:subClassOf dbpo:Museum}
  UNION
  {?s a yago:Museum103800563}
  UNION
  {?s a ?t . ?t rdfs:subClassOf yago:Museum103800563}
  OPTIONAL { ?s geo:lat ?lat}
  OPTIONAL { ?s geo:long ?lon}
  OPTIONAL { ?s geo:geometry ?geom}
  OPTIONAL { ?s grs:point ?point}
}
```

Note that the query uses both YAGO (Suchanek et al. 2007) and DBpedia ontology parent classes and retrieves coordinates, if present, in whatever form they may be stored.

The resulting data for inclusion in the local index, which includes the URI that contains the place name, are stored similarly to the local geo-data, within a PostGIS database. For the purposes of our experimental system the DBpedia coordinates were converted to the UK National Grid map projection (i.e. in metres) to match the native coordinates of the geo-spatial data. In order to scale the system to work globally all coordinates could be geodetic (latitude and longitude), which systems such as PostGIS can support for purposes of spatial query.

Our system is designed to maintain links between DBpedia objects and corresponding features in reference geo-data. These links are required for automated processing of queries where SPARQL queries are required to determine properties of DBpedia instances that are represented in the spatial aspect of the query by reference geometry. This occurs for example when querying the properties of linear or area objects that hold a specified spatial relationship to some other named feature. Currently implemented methods for matching are similar to those described by Stadler et al (2011) in their work on linking OSM to DBpedia, being a combination of name matching and distance between to the two geometries, of which the DBpedia

geometry will always be a point. In our method, place names are normalised before performing an exact match, as the use of methods such as Levenshtein edit distance was found to result in too many false positives. These methods will continue to be refined through exploitation of additional evidence such as feature type.

4 Example Queries and Experimental Results

In this section we present examples of the different sorts of query that can be processed using the various methods that we employ, specifically for proximity from lines and areas, crossing (overlap) and containment. The method of combining qualitative DBpedia properties that infer containment with quantitative methods based on reference geo-data are examined in more detail than the others in order to reveal the balance between the numbers of results produced by the two containment methods.

4.1 Non-spatial Queries to DBpedia

Here the user can specify a property, selected from a drop down menu, and a target location. To find such properties for the query “Find the Capital of the United Kingdom”, a simple SPARQL query is constructed as follows,

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX dbpo:   <http://dbpedia.org/ontology/>
SELECT ?o WHERE {dbpedia:$s dbpo:$p ?o}
```

where to \$s and \$p are substituted by 'United_Kingdom' and 'capital', respectively.

4.2 Proximity Queries

The use of detailed geo-data enables proximity queries on DBpedia geographic instances to be performed relative to point, line and area features representing geographical features named by the user in the query. In this procedure the reference feature named in the query is represented by the reference geo-data, while the subject of the query is represented geometrically either by the point coordinate geometry of georeferenced DBpedia features or reference geo-data geometry that has been matched to the DBpedia instance. Here will illustrate examples of both situations.

For the query “Find churches within 1km of the River Thames” (i.e. return the references to the relevant DBpedia instances), the user’s query term “River Thames” is represented by geometry from the reference geo-data, while the locations of the churches are those from DBpedia, found here via the local spatial index of DBpedia instances. Figure 2 illustrates a map of the results with some of the retrieved instances listed.

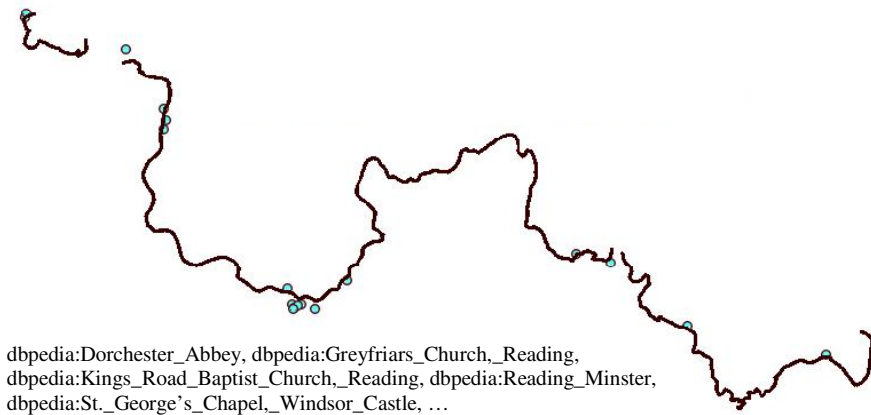


Fig. 2. Some results for a query to find DBpedia churches within 1km of the River Thames. Geometry data courtesy UK Ordnance Survey.

Another example of using point geometry from DBpedia in combination with reference geometry is for the query “Find hospitals outside and within 10km of the city of Cardiff”. Here the polygonal boundary of Cardiff is obtained from the reference geo-data and the point locations of the hospitals are obtained from the spatial index of DBpedia. The spatial query with PostGIS uses a combination of a distance constraint and a negation of inside in order to obtain locations that are outside the city. Figure 3 illustrates the results.



Fig. 3. Results for query for DBpedia hospitals outside and within 10km of the city of Cardiff. Geometry data courtesy UK Ordnance Survey.

4.3 Crossing Queries

We illustrate the use of reference geo-data to represent the retrieved DBpedia instances with the query “Find the mouths of the rivers that cross Oxford”. Here the PostGIS spatial database is used to find rivers that satisfy the spatial constraint, where the geometry of the rivers and the city boundary come from the reference geometry, while the corresponding DBpedia instances, that match the river names, are queried with SPARQL to find the mouths of the rivers. Automation of this query requires links between reference geo-data features and corresponding DBpedia instances as explained in Section 3.

River	Mouth
"Oxford Canal",	dbpedia:River_Thames,
"River Cherwell",	dbpedia:North_Sea,
"River Thames or Isis"	dbpedia:Thames_Estuary



Fig. 4. Results for query to find the mouths of the rivers that cross Oxford. Geometry data courtesy UK Ordnance Survey.

4.4 Containment Queries

Containment queries relative to named regions with known boundaries can be performed by a combination of quantitative and qualitative methods and, as indicated previously, in the absence of full quantitative geo-referencing of geographical DBpedia instances both methods are required to maximise the completeness of the response. In order to gain some insight into the balance between the use of coordinates and of properties that imply spatial containment, we selected ten UK cities (Bath, Birmingham, Bristol, Cardiff, Durham, Leeds, Liverpool, Manchester, Newport (South Wales), Nottingham) as the target of a set of queries to retrieve instances of the following eight feature types: Churches, Historic Buildings, Hospitals, Hotels, Libraries, Museums, Shopping Malls, and Stadiums. Spatial queries used the DBpedia coordinates to determine containment in city boundaries obtained from the reference geo-data, while qualitative containment was determined as explained previously, using all properties that have the respective city as object, subject to filtering via the predetermined list of non-containment properties. The containment properties (after filtering) that were used in this study are listed below:

Containment = {dbpo:district, dbpo:homeport, dbpo:location, dbpo:locationCity, dbpo:municipality, dbpo:owner, dbpo:principalArea, dbpo:region, dbpp:district, dbpp:city, dbpp:location, dbpp:locationTown, dbpp:mapCaption, dbpp:municipality, dbpp:owner, dbpp:parish, dbpp:postTown, dbpp:region}

Note that some of the results from the second method could duplicate instances in the local spatial index, where a DBpedia instance has both coordinates and spatial containment properties. Table 1 presents a summary of the results in which we report, for each place type, the numbers of instances found within the respective containing city using spatial containment relations (S) and geographic coordinates (G). For a set of instances retrieved using spatial containment relations (S), the sets SG, SG_i and SG_n are constructed by filtering the proportion of S with geographic coordinates (SG), then using the coordinates to test those which fall within the boundaries of the reference area (SG_i) and those which do not (SG_n).

Table 1. Per feature and per settlement area, number of features related to the area with spatial containment relations (S), those of S with geographic coordinates (SG), those of SG that fall within the boundaries of the reference area (SGI), those and that do not (SGN). G represents all features with geographic coordinates which fall within the boundary.

	Bath					Birmingham					Bristol					Cardiff					Liverpool				
	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G
churches	9	7	7	-	8	3	3	3	-	18	35	2	2	-	2	2	2	-	4	22	18	18	-	38	
historicbuildings	3	3	3	-	3	2	1	1	-	28	-	-	-	-	-	1	1	1	-	3	16	16	16	-	22
hospitals	2	1	1	-	1	4	3	3	-	13	10	7	6	1	9	3	3	3	-	4	3	3	3	-	6
hotels	1	1	-	1	1	3	3	3	-	7	1	1	-	1	-	2	2	2	-	3	3	3	3	-	9
libraries	-	-	-	-	-	2	2	2	-	3	1	1	1	-	1	-	-	-	-	1	-	-	-	-	1
museums	13	11	9	2	9	7	6	6	-	24	17	11	10	1	8	8	7	4	3	4	10	9	4	5	9
shoppingmalls	1	1	1	-	1	3	2	2	-	11	3	3	3	-	3	5	5	5	-	5	5	4	4	-	5
stadiums	2	2	2	-	2	10	9	8	1	13	2	2	2	-	2	12	12	12	-	12	4	4	3	1	6
theatres	5	5	5	-	5	6	6	6	-	13	5	3	3	-	5	3	2	2	-	5	3	3	3	-	7
Total:	36	31	28	3	30	40	35	34	1	130	74	30	27	3	30	36	34	31	3	41	66	60	54	6	103

	Durham					Newport					Leeds					Manchester					Nottingham				
	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G	S	SG	SGi	SGn	G
churches	-	-	-	-	2	1	1	1	-	1	6	5	4	1	7	2	2	2	-	33	14	13	13	-	46
historicbuildings	-	-	-	-	-	1	1	1	-	1	9	9	7	2	8	2	2	2	-	25	16	16	16	-	53
hospitals	3	2	2	-	2	2	2	1	1	1	5	5	5	-	5	5	5	5	-	18	3	1	1	-	2
hotels	-	-	-	-	-	1	1	1	-	1	2	2	2	-	4	5	2	2	-	6	-	-	-	-	-
libraries	1	1	1	-	1	-	-	-	-	-	1	1	1	-	2	2	1	1	-	6	-	-	-	-	-
museums	1	1	-	1	3	-	-	-	-	2	1	1	1	-	7	8	7	6	1	19	3	2	2	-	2
shoppingmalls	2	2	1	1	1	2	2	2	-	2	4	2	2	-	5	4	2	2	-	10	-	-	-	-	2
stadiums	-	-	-	-	-	4	4	3	1	3	4	4	4	-	4	9	8	8	-	27	4	3	3	-	3
theatres	-	-	-	-	-	-	-	-	-	1	2	2	2	-	2	13	7	7	-	16	4	3	3	-	3
Total:	7	6	4	2	9	11	11	8	3	12	34	31	28	3	44	50	36	35	1	160	44	38	38	-	111

Table 2. Totals for each category for all cities and associated percentages. $|G/(G+S-SG)|$ are the instances retrieved using spatial coordinates and geographic boundary only, $|SG/G|$ the percentage of instances with coordinates that also have spatial relation properties. $|SG/S|$ the percentage of instances with spatial relations that also have coordinates. $|SGn/SGi|$ the number of features that are linked by containment to the area but are not in the area according to our geo-data reference city boundary, and $|SGi/G|$ the percentage of instances with coordinates that also have spatial relations and are within the city boundary.

	TOTALS										
	S	SG	SGi	SGn	G	G+S	G/(G+S)	SG/G	SG/S	SGn/S	SGi/G
churches	94	53	52	1	159	200	79.5%	33.3%	56.4%	1.9%	32.7%
historicbuildings	50	49	47	2	143	144	99.3%	34.3%	98.0%	4.1%	32.9%
hospitals	40	32	30	2	61	69	88.4%	52.5%	80.0%	6.3%	49.2%
hotels	18	15	12	3	31	34	91.2%	48.4%	83.3%	20.0%	38.7%
libraries	7	6	6	-	15	16	93.8%	40.0%	85.7%	0.0%	40.0%
museums	68	55	42	13	87	100	87.0%	63.2%	80.9%	23.6%	48.3%
shoppingmalls	29	23	22	1	45	51	88.2%	51.1%	79.3%	4.3%	48.9%
stadiums	51	48	45	3	72	75	96.0%	66.7%	94.1%	6.3%	62.5%
theatres	41	31	31	-	57	67	85.1%	54.4%	75.6%	0.0%	54.4%
Total:	398	312	287	25	670	756	88.6%	46.6%	78.4%	8.0%	42.8%

It may be noted that there is considerable disparity in the proportion of instances of particular types that are found only due to coordinates and those found only due to qualitative properties, where the number of results using geographic coordinates are nearly always higher in this analysis (but for a notable exception see Bristol in Table 1). Using only geographic coordinates and a geographic footprint retrieves 88.6% of all results, the other 11.4% being provided by instances without geographic coordinates but with qualitative relations to the containment instance. When compared with pure SPARQL queries that use only qualitative relations, our method almost doubles, on average, the number of instances retrieved, i.e. 398 against 756. These results can therefore be regarded as providing strong validation for the benefits of combining both methods.

5 Conclusions and Future Work

We have presented the design of an experimental system designed to demonstrate hybrid methods for performing spatial query on Semantic Web resources such as DBpedia, with the intention to maximise the completeness of the answers with respect to finding relevant content and obeying user specified spatial operators. The approach is novel with regard to the combination of exploiting high quality geo-data and mixing quantitative and qualitative methods to obtain results. In addition to providing examples of how the high quality geo-data can be employed to find results based on a variety of spatial relations, we have demonstrated that, for the case of containment queries, the combination of quantitative geo-spatial query with qualitative query produces greatly superior results to the use of these methods in isolation. The work presented here is only concerned with the quality of the results. Performance issues, particularly timing, will be addressed in future work.

It may be noted that, even with the methods presented, some spatial instances will be omitted from the results, where these instances do not have either coordinates or explicit or implicit spatial property relations. Future work will focus on geo-referencing of these instances to enable them to be accessible via geo-spatial query. The present system employs a simple text-based user interface. This will be enhanced with map-based feedback. Reference geo-data will be extended to include resources such as OSM and Geonames and hence gain international coverage and improved disambiguation facilities. Our approach is intended to scale up to support access to multiple Semantic Web resources that provide geographical information. Thus the local index will be extended to include reference to other geographically informative RDF resources, which will be linked to reference geo-data, using similarity matching methods.

References

1. Buscaldi, D.: Resource Integration for Question Answering and Geographical Information Retrieval. Research project report, Polytechnic University of Valencia Valencia, Spain: The Department of Information Systems and Computation (2007), <http://users.dsic.upv.es/~proso/resources/BuscaldiIDEA.pdf>

2. Battle, R., Kolas, D.: GeoSPARQL: Enabling a Geospatial Semantic Web. Submitted Semantic Web Journal (2011), http://www.semantic-web-journal.net/sites/default/files/swj176_1.pdf
3. Brodt, A., Nicklas, D., Mitschang, B.: Deep integration of spatial query processing into native RDF triple stores. In: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (2010)
4. Della Valle, E., Qasim, H.M., Celino, I.: Towards Treating GIS as Virtual RDF Graphs. In: Proceedings of 1st International Workshop on Pervasive Web Mapping, Geoprocessing and Services (WebMGS (2010)
5. De Rouck, C., Van Laere, O., Schockaert, S., Dhoedt, B.: Georeferencing Wikipedia pages using language models from Flickr. In: Proceedings of the Terra Cognita Workshop (ISWC), pp. 3–10 (2011)
6. Hackley, M.: How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets. *Environment and Planning B*, 682–703 (2010)
7. Hahmann, S., Burghardt, D.: Connecting LinkedGeoData and Geonames in the Spatial Semantic Web. In: 6th International GIScience Conference (2010)
8. Hartrumpf, S., Leveling, J.: Recursive Question Decomposition for Answering Complex Geographic Questions. In: Peters, C., Di Nunzio, G.M., Kurimo, M., Mandl, T., Mostefa, D., Peñas, A., Roda, G. (eds.) CLEF 2009, Part I. LNCS, vol. 6241, pp. 310–317. Springer, Heidelberg (2010)
9. Katz, B., Lin, J.J.: Start and Beyond. In: Proceedings of 6th World Multiconference on Systemics, Cybernetics, and Informatics (2002)
10. Lin, J., Katz, B.: Question Answering Techniques for the World Wide Web. In: 11th Conference of European Association of Computational Linguistics, EACL 2003 (2003)
11. Luque, J., et al.: GeoVAQA: a voice activated geographical question answering system. IV Jornadas en Tecnología del Halba (2008), http://jth2006.unizar.es/finals/4jth_158.pdf
12. Mishra, A., Mishra, N., Agrawal, A.: Context-aware restricted geographical domain question answering system. In: International Conference on Computational Intelligence and Communication Networks, CICN 2010, IEEE, Washington (2010)
13. Mollá, D., Vicedo, J.L.: Question Answering in Restricted Domains: An Overview. *Computational Linguistics* 33(1) (2007)
14. Mooney, P., Corcoran, P., Winstanley, A.C.: Towards quality metrics for OpenStreetMap. In: Proceedings of the 18th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (2010)
15. Purves, R.S., Clough, P., Jones, C.B., et al.: The design and implementation of SPIRIT : a spatially-aware search engine for information retrieval on the internet. *International Journal of Geographical Information Systems* 21(7), 717–745 (2007)
16. Salas, J.M., Harth, A.: Finding spatial equivalences across multiple RDF datasets. In: Terra Cognita 2011, co-located with ISWC, Bonn, Germany (2011)
17. Stadler, C., Lehmann, J., Höffner, K., Auer, S.: LinkedGeoData: A core for a web of spatial open data. To appear in *Semantic Web Journal* (in press), http://svn.aksw.org/papers/2011/SWJ_LinkedGeoData/public.pdf
18. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. In: Proceedings of the 16th International Conference on World Wide Web, pp. 697–706 (2007)
19. Waldinger, R., Appelt, D.E., Fry, J., Israel, D.J., Jarvis, P., Martin, D., Riehemann, S., Stickel, M.E., Tyson, M., Hobbs, J., Dungan, J.L.: Deductive Question Answering from Multiple Resources. In: *New Directions in Question Answering*. AAAI (2004), <http://www.ai.sri.com/pubs/files/986.pdf>

Extracting Dynamic Urban Mobility Patterns from Mobile Phone Data

Yihong Yuan^{1,2} and Martin Raubal¹

¹ Institute of Cartography and Geoinformation, ETH Zurich, 8093 Zurich, Switzerland

² Department of Geography, University of California, Santa Barbara, CA, 93106, USA
yuan@geog.ucsb.edu, mraubal@ethz.ch

Abstract. The rapid development of information and communication technologies (ICTs) has provided rich resources for spatio-temporal data mining and knowledge discovery in modern societies. Previous research has focused on understanding aggregated urban mobility patterns based on mobile phone datasets, such as extracting activity hotspots and clusters. In this paper, we aim to go one step further from identifying aggregated mobility patterns. Using hourly time series we extract and represent the *dynamic mobility patterns* in different urban areas. A Dynamic Time Warping (DTW) algorithm is applied to measure the similarity between these time series, which also provides input for classifying different urban areas based on their mobility patterns. In addition, we investigate the outlier urban areas identified through abnormal mobility patterns. The results can be utilized by researchers and policy makers to understand the dynamic nature of different urban areas, as well as updating environmental and transportation policies.

Keywords: Mobile phone datasets, Urban mobility patterns, Dynamic Time Warping, Time series.

1 Introduction

Identifying urban mobility patterns has been a continuing research topic in GIScience, transportation planning, and behavior modeling. Since the time-dimension is considered an important factor for most social activities, understanding the dynamics of the daily mobility patterns is essential for the management and planning of urban facilities and services [1, 2]. However, most of the previous research in this field is based on data acquired from travel diaries and questionnaires, which is a widely adopted data collection method when studying individual travel behavior [3]. Due to the limited number of people covered by travel diaries, these datasets fail to provide comprehensive evidence when studying the characteristics of the whole urban system, such as identifying clusters of urban mobility.

Meanwhile, the development of information and communication technologies (ICTs) has created a wide range of new spatio-temporal data sources (e.g., georeferenced mobile phone records), leading to research that focuses on characterizing urban mobility patterns from mobile phone datasets (e.g., the real-time Rome project at the MIT SENSEable City

Lab¹). Undoubtedly, mobile phone datasets opened the way to a new paradigm in urban planning, i.e., Real-time cities [4], as well as facilitating studies on behavior analysis and spatio-temporal data mining [5]. Researchers believe that urban structure has a strong impact on urban-scale mobility patterns, indicating that different areas inside a city are associated with different inhabitants' motion patterns [6, 7]; therefore, previous research has focused on extracting aggregated patterns in different urban areas from mobile phone data, such as hotspots, clusters, and points of interest (POIs) [8]. However, there has not been sufficient research on characterizing and classifying mobility patterns in different urban areas from a *dynamic perspective*, i.e., analyzing these patterns with respect to time. Although the extraction of aggregated patterns (i.e., hotspots and clusters) offers valuable input for maintaining the sustainability of urban mobility, it fails to provide sufficient information for understanding the "rhythm" of an urban system. The objective of this research is to go a step beyond the aggregation of individual mobility. We analyze the hourly patterns (time series) of mobility aggregation in different urban areas and demonstrate their differences. For instance, time series associated with a central business district (CBD) would be different from suburban areas. Exploring these patterns will be helpful for policy makers in understanding the dynamic nature of different urban areas, as well as updating environmental and transportation policies. Moreover, the methodology can also be applied to identify abnormal mobility patterns in some special districts, for example, a high crime rate area.

The analysis in this research is based on a mobile phone dataset from northeast China. We will measure the similarity of different urban areas based on a Dynamic Time Warping algorithm (DTW): this is a well-developed algorithm in the field of speech recognition and signal processing for matching two time series, but it has rarely been used for urban mobility modeling [9]. Next, we will classify the time series based on hierarchical clustering, which allows for the detection of outlier urban patterns. The results can also be used as a reference for residents' activities, including long-term choices such as where to live, and short-term choices such as daily activity scheduling.

The remainder of this paper is organized as follows: Section 2 describes related work in the areas of mobility modeling, mobile phone data analysis, and Dynamic Time Warping. Section 3 introduces the basic research design, including the description of the dataset and the methodology. Section 4 presents the data analysis, and we conclude this research in Section 5.

2 Related Work

2.1 Mobility Modeling and Mobile Phone Data

Modeling human mobility patterns has become an important research question in various fields such as Geographic Information Science, Transportation, and Physics. Much progress has been made regarding the theories, methodologies, and applications. Larsen [10] identified five types of mobility: 1) Physical travel of people (e.g., work, leisure,

¹ <http://senseable.mit.edu/realtimerome/>

family life); 2) Physical travel of objects (e.g., products to customers); 3) Imagination travel (e.g., memories, books, movies); 4) Visual travel (e.g., internet surfing on Google Earth); and 5) Communication travel (e.g., person-to-person messages via telephones, letters, emails, etc.). In this research, when referring to “human mobility” we mainly focus on characterizing the 1st category of human mobility (Physical travel of people).

Due to the widespread usage of mobile phones, several studies have been conducted with a focus on extracting the characteristics of human mobility from georeferenced mobile phone data [11]. Since individuals are atoms in an urban system, the spatio-temporal characteristics of an urban system could be viewed as a generalization of individual behavior; therefore, mobile phone data also provide new insights in analyzing the aggregated mobility patterns of phone users in urban systems. Researchers have identified two major perspectives when exploring human mobility patterns from mobile phone data [12]:

- (a) **Individual perspective:** This category of research mainly focuses on identifying individual trajectory patterns, which is related to the theme of pattern recognition in Physics and Computer Science. For example, Gonzalez et al. [13] studied the individual trajectories of 100,000 mobile phone users based on tracked location data for over six months, providing new input to understanding the basic laws of human motion. Song et al. [14] examined the regularity of human trajectories based on mobile phone data, and their results indicate that human mobility is highly predictive. Some researchers have combined the location information with social attributes of the phone users, such as with the social positioning method (SPM) [15]. Since the usage of mobile phones can affect the mobility patterns of their users, previous studies have also focused on the interaction between ICTs and human activity-travel behavior [16, 17].
- (b) **Urban perspective:** Cities can be considered complex systems that are constituted by different processes and elements [2]. The rapid development of ICTs not only provides a rich data source for modeling urban systems, but also resulted in inevitable changes in the spatio-temporal characteristics of urban mobility. Researchers have focused on the following two aspects when studying the development in urban and regional planning based on mobile phone data:
 - (i) **Spatial division and morphology:** For example, Kang et al. [18] investigated how patterns of human mobility inside cities are affected by two urban morphological characteristics, i.e., compactness and size.
 - (ii) **Spatial clustering and spread:** The study of hotspot clustering patterns has been addressed in many studies. In the real-time Rome project conducted by the MIT SENSEable City Lab, researchers studied the congregation of tourists and the gathering of people during special events². Another similar project is “Mobile Landscape Graz in Real Time”, which concentrates on the activity distribution of phone users in the city of Graz, Austria³.

² <http://senseable.mit.edu/realtimerome/>

³ <http://senseable.mit.edu/graz/>

The analysis in this research is conducted from the urban perspective. As stated in Section 1, most previous research has concentrated on exploring aggregated patterns when analyzing urban mobility from mobile phone datasets. Here we focus on the temporal patterns of urban mobility. We use DTW to characterize and classify the mobility time series associated with different urban areas, which extends previous research on spatial clustering and mobility spread. The DTW algorithm has been identified as one of the most useful methods to measure the similarity between two time series, which minimizes the effects of shifting and distortion in time [19]. Section 2.2 provides the background of DTW and its applications.

2.2 Dynamic Time Warping and Its Applications

One important research question regarding time series data is finding whether two time series represent similar behavior [20]. Traditional distance measures, such as Euclidean distance, are not suitable for measuring the distance between time series data. For example, consider two time series $A[1,1,1,1,2,10,1,1,1]$ and $B[1,1,1,1,10,2,1,1,1]$: the Euclidean distance between A and B is $\sqrt{128}$. This is a fairly large number, which implies dissimilarity between the two given time series; however, the structures of the two series are actually very similar to each other. Therefore, researchers started to look for new algorithms to measure the similarity between two time series. Moreover, in the fields of Computer Science and Mathematics, researchers also used Discrete Fréchet Distance to measure the similarity between two curves [21]; however, this method is very sensitive to outliers and displacements [22], therefore it is not very appropriate for time series data. Here, Dynamic Time Warping (DTW) is proposed to find an optimal match between two given time-dependent sequences [23]. This algorithm has been well developed to measure the similarity between time series in various research areas, such as speech recognition, motion detection, or signal processing [24]. DTW has also been used for analyzing human trajectories and motion patterns, for example, Lee et al. [25] utilized DTW to classify the trajectories of moving objects.

Fig.1 represents the process of calculating the DTW distance between two example time series. First a DTW grid is constructed. Inside each grid cell a distance measure is applied to compare the corresponding elements (here we use absolute differences) of the two time series. In order to find the best match between these two sequences, one needs to find a path through the grid which minimizes the total distance; this is considered the DTW distance between the two series.

The biggest advantage of DTW is that one can obtain a robust time alignment between reference and test patterns with a high tolerance of element displacement [26]. It can also match series with different lengths, which is very useful for some applications such as handwriting recognition. However, sometimes DTW tends to over-distort the series to create an unrealistic correspondence between elements; therefore, it is applicable to set local constraints and global constraints on the path. This prevents very short features matching with very long ones [19].

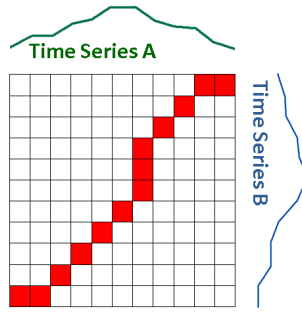


Fig. 1. DTW algorithm

As mentioned in Section 1, although DTW has been applied to analyze individual trajectory patterns, only a few studies have utilized this method to explore urban-scale patterns, most of which concentrate on remote sensing data [27, 28]. Researchers have proposed several other methodologies to compare two time series, such as Longest Common Subsequence (LCSS), but DTW has a high performance for series in which the same classes are best characterized by their shapes rather than their values. In this research we focus on the internal structure of the mobility time series instead of their magnitude. Since DTW can be used to warp the time series, it allows us to group similar mobility patterns together, even though the corresponding elements in the two series are not exactly aligned with each other (see example in Section 3.2). More specifically, here we will use DTW to measure the similarity of hourly population density trends of different urban areas. The results of the similarity measure will serve as the basis for urban classification and outlier detection. In addition, we will discuss the issue of comparing the mobility pattern of a reference area, i.e., a benchmark, to other urban areas.

3 Research Design

3.1 Dataset

The analysis is based on a dataset from city A⁴ (acquired from a major mobile phone operator in China), which is a commercial and transportation center in northeast China. The dataset covers approximately one million mobile phone users (20% of the city population) and includes mobile phone connection records for a time span of 9 days (4 weekend days and 5 weekdays). It includes the time, duration, and approximate location of mobile phone connections, as well as the age and gender attributes of the users. Table 1 provides a sample record. The phone number, longitude, and latitude are not shown for privacy reasons. For each user, the location of the nearest mobile phone base tower is recorded both when the user makes and receives a phone call, resulting in a positional data accuracy of about 300m-500m.

⁴ The name of the city is not shown as requested by the data provider.

Table 1. Sample record from the dataset

Phone number	Longitude	Latitude	Time	Duration
1360*****	126.*****	45.*****	12:06:12	5mins

3.2 Methodology

As discussed in Section 2, we use DTW to measure the similarity of hourly mobility patterns between different urban areas. This algorithm allows us to group similar patterns together, as well as identifying outlier patterns. Due to the complexity of urban systems, it is highly possible that similar mobility patterns may have various forms in terms of their time dimensions. Fig. 2a shows two example series that are similar (series 2 is created from series 1 using the lag operator lag=1, the y axis of both figures are normalized to [0, 1] for simplicity). Both series have two peak time periods (one in the morning, the other in the afternoon). For comparison, Fig. 2b shows two series that are highly distinct from each other (series 3 is a flat series).

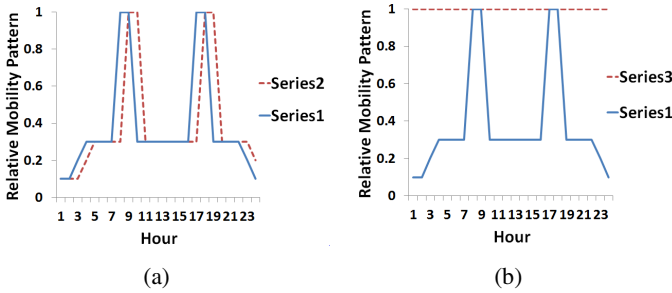


Fig. 2. Example series: (a) Two similar patterns; (b) Two distinct patterns

The distances measured by DTW, Euclidean distance and the Discrete Fréchet Distance are presented in Table 2.

Table 2. DTW, Euclidean and Discrete Fréchet distance for example series

	Dis1 (Series 1 vs Series 2)	Dis2 (Series 1 vs Series 3)	Distance Ratio (Dis2/Dis1)
DTW	0.00208	0.31	149.04
Euclidean	1.41	3.33	2.36
Fréchet	0.70	0.90	1.29

As can be seen, the distance ratio indicates that DTW shows a much better performance of distinguishing different time series than the other two methods; therefore, it is a more useful method for researchers to quantify the similarity of dynamic mobility patterns.

In this research, the data analysis will be conducted in the following three steps:

3.2.1 Summarize Dynamic Population from Cell Phone Records

To summarize the dynamic mobility patterns in different urban areas, we first need to divide the study area into sub-areas. One option is to divide the study area into grid cells [18]; however, it is difficult to decide on the appropriate cell size. Moreover, it is highly possible that the number of base towers in each cell varies, resulting in higher mobility in areas with higher tower density. Therefore, we decided to divide the study area into Voronoi polygons based on the spatial distribution of cell phone towers (Fig. 3), and then to summarize the hourly phone call frequencies for each polygon. Each Voronoi polygon is associated with a time series to represent its hourly phone call frequency pattern. To further extract the number of people (i.e., active mobile phone users) in each cell, we eliminated the repeated phone calls made by the same user.

Note that all the numbers here are on an average daily basis. To normalize the results, each population count is divided by the size of the given polygon. Since the analysis for large polygons has relatively low spatial accuracy resulting from the low density of base towers in the surrounding area, we only perform the analysis for polygons smaller than 10 km². As indicated in Fig. 3, these polygons (highlighted) cover the majority of the downtown area.

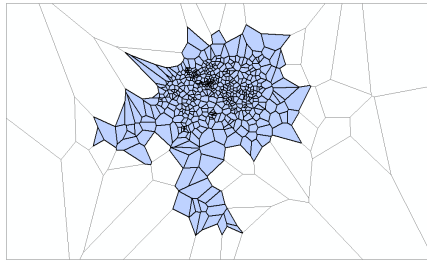


Fig. 3. Voronoi polygons smaller than 10km²

Last, we calculate relative mobility patterns for each polygon. In relative time series, for each cell its values are divided by the maximum of the 24 hourly values. This standardizes the magnitude of data and also helps in further investigating the internal structure of each time series. Since the main focus of this research is not on the absolute value of each series, we use relative time series instead of the original ones to measure the similarity of mobility patterns between polygons.

3.2.2 Calculate DTW Distance Matrix

Based on the algorithm described in Section 2, we construct the DTW distance matrix for the relative time series associated with each of the selected Voronoi polygons. The output is a distance matrix D , in which D_{ij} represents the DTW distance between cell polygon i and j . We use a global constraint “Sakoe-Chiba band”, which has a fixed windows width in both horizontal and vertical directions [23]. Here the window size is set to be 4, indicating that the maximum allowable absolute time deviation between two matched elements is 4 hours. This constraint helps to prevent unrealistic distortion in the time dimension, such as matching the evening hour patterns with morning patterns.

3.2.3 Analyze Urban Mobility Patterns Based on DTW Distance Matrix

Based on the DTW matrix, one can explore the dynamic patterns of urban areas from various perspectives, either addressing the “similarity” or “dissimilarity” of urban divisions. In this paper we will conduct two example analyses for both circumstances based on the distance matrix constructed in step 2. The first one focuses on mapping the mobility similarity to reference areas, whereas the second example concentrates on detecting outlier patterns. Note that to further clean up the data, polygons with zero-phone call frequencies are eliminated. The analysis is presented in detail in Section 4.

4 Data Analysis

4.1 Mapping the Similarity to Reference Areas

In urban studies, it is common for researchers to select one or more particular areas as case studies for data collection and analysis. Many of these studies are related to human mobility patterns, such as crime trends, traffic congestion, etc. Although there are usually many other control variables in the analysis, identifying the mobility similarity between a selected area (reference area) and other areas can provide references for further analysis.

Fig. 4 represents the similarity measure of mobility patterns between a reference polygon (marked red, where a major commercial street is located) and other urban areas. Dark brown color indicates a more similar mobility pattern (shorter DTW distance), whereas the light yellow color indicates a less similar one. As can be seen from Fig. 4, the average DTW distance on weekdays ($2.73e-2$) appears to be slightly smaller compared to that on weekends ($2.85e-2$) based on a paired two sample t test ($p < 0.001$), indicating that the mobility patterns on weekdays are closer to the pattern in the reference area. A potential reason is that most human social activities during weekends (i.e., grocery shopping, leisure activities) do not have such strict time constraints as the ones on weekdays (i.e., go to school / work), so it is highly possible that there are more irregular patterns during weekends (further confirmed in the outlier analysis in Section 4.2). In addition, it appears that the polygons surrounding the commercial street show a more similar pattern to the reference area on weekends than on weekdays (see the zoomed-in subfigures of Fig. 4a, b), indicating a potential mobility correlation among those areas during weekends. This also represents the opposite of the general trend of the whole study area, where mobility on weekdays is closer to the pattern in the reference area. This indicates that spatial scale plays an important role in this analysis. However, in order to generate further conclusions for other urban study questions (e.g., traffic congestion), we will need additional socio-economic data to conduct additional correlation analyses. Fig. 4 is only a first step of measuring the similarity between different urban areas in terms of dynamic mobility patterns, and it provides an initial reference for socio-economic studies.

One can also define the reference (benchmark) series manually. For example, we define the benchmark series as $[1, 1]$, representing an evenly distributed mobility pattern during both day and night hours.

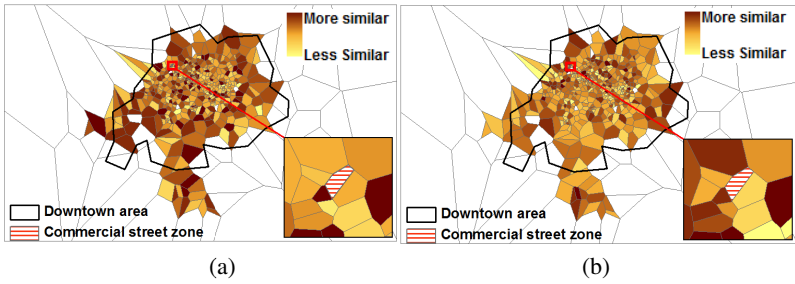


Fig. 4. Mapping the DTW distance between reference area and other areas. (a) Weekdays; (b) Weekends

Fig. 5 shows the distribution of DTW distances between the benchmark series and the study areas. This method is very useful for interpreting the internal structure of dynamic mobility patterns for a particular cell polygon. In this case, polygons with a smaller DTW distance have more evenly distributed mobility patterns.

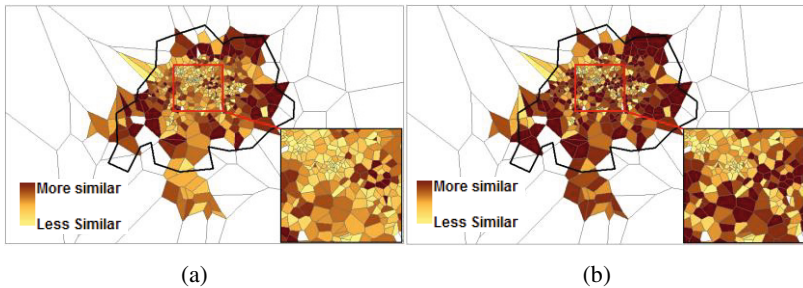


Fig. 5. Mapping the DTW distance between a benchmark series and other series. (a) Weekdays (b) Weekends

As can also be seen from the histogram (Fig. 6), in the first three groups (DTW distance < 0.2), there are more polygons on weekends than on weekdays, indicating that the mobility patterns on weekends are closer to an evenly distributed pattern. This is consistent with common sense that activities during weekends have less time constraints.

Similarly, we can define other benchmark series. For example: [0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0] represents a pattern that expresses the fact that there is only one peak during the day. Note that the numbers in benchmark series can be any value between 0 and 1, and it is not necessary to use binary values. The principle here is similar to the studies utilizing DTW to detect a particular handwriting style or speech tone pattern. By matching pre-defined benchmark series with study areas, we can investigate various patterns that are of interest.

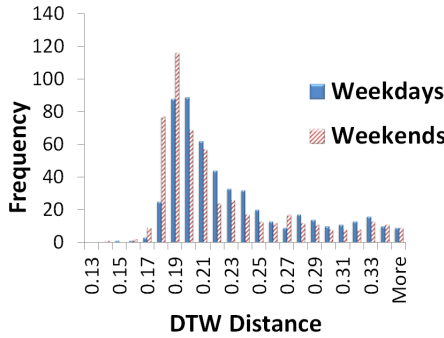


Fig. 6. Histogram of DTW distance for weekdays and weekends

4.2 Outlier Detection

As discussed in [29], outlier mining techniques can be used to investigate abnormal activities such as traffic accidents. From a broader perspective, since urban-scale mobility patterns are strongly affected by the urban structures, identifying abnormal mobility patterns can be helpful for researchers and policy makers to investigate the functioning patterns of different urban areas, as well as optimizing the distribution of urban services (e.g., Police patrol). Moreover, this technique can also be applied to detect potential incidents by comparing given patterns in a certain area to its regular pattern. Therefore, in the second analysis we explore outlier detection based on the DTW distance matrix discussed in Section 3. Our objective is to identify cell polygons with abnormal mobility patterns. Since hierarchical classification can operate directly on the distance matrix, we adopt this method to classify the mobility time series. The algorithm is defined in Fig.7.

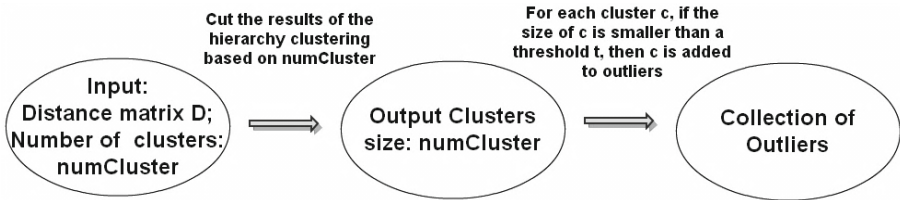


Fig. 7. Outlier detection algorithm

There are several methods to set the number of clusters in hierarchical classification; however, this value is often affected by specific application scenarios. As an example analysis, here we adopt the criteria discussed in [30], where $numCluster = \max(2; \sqrt{n/2})$, n is the number of entries, and threshold t is defined as 3.

In the classification, we detected 15 outliers for weekdays and 18 for weekends. All the other cells are aggregated into one class. To further investigate the structures of the outlier series, we first define what a typical “normal series” looks like. Fig. 8

shows an average series for both weekdays and weekends after removing the outlier polygons. As can be seen, a normal series has two mobility peaks each day: one is around 9am; the other is around 6pm. The mobility density reaches the lowest point between 2-4am. This is consistent with common sense. On weekends the mobility density is slightly higher during night hours in this case, but there is no substantial difference between weekdays and weekends regarding the average patterns.

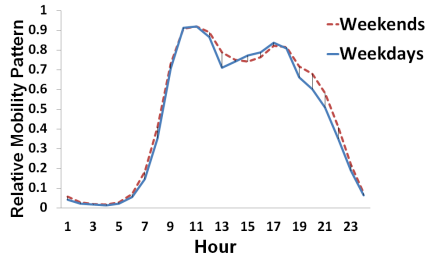


Fig. 8. Average normal series

Fig. 9 shows the results of the outlier detection. The detected outliers are marked red, other cell polygons are marked light blue. We can see that there are slight differences between weekdays and weekends.

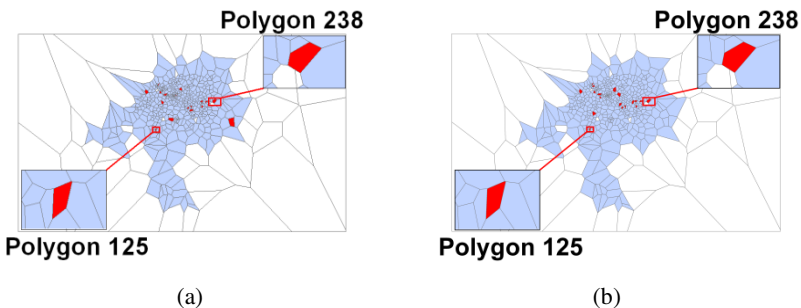


Fig. 9. Outlier polygons. (a) Weekdays; (b) Weekends

As a comparison, Fig. 10 shows two example outlier time series (zoomed-in polygons in Fig. 9). Referring back to the landmarks on Google Map⁵, the plot leads us to the following hypothesis to explain the abnormality of the areas:

In polygon 238 there are many night clubs and other leisure facilities for night hours. This may explain the abnormal high density rate after midnight. Since there is a big international trade center only open during weekdays, this possibly explains why the mobility during daytime is not consistent with regular work hours on weekends.

⁵ <http://maps.google.com>; The map with landmarks is not shown as required by the data provider.

In polygon 125 there are several community colleges and training schools. There are not many night clubs in this area. The mobility density continues to be high between 8am and 8pm on weekends, indicating a noticeable difference in mobility patterns between weekdays and weekends.

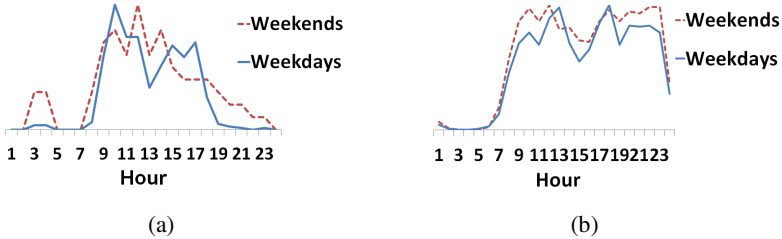


Fig. 10. Outlier patterns. (a) Polygon 238; (b) Polygon 125

Additional information is needed to test the above hypothesis, which is not the focus of this research. Generally, the above provides us with a novel method of detecting the abnormality of urban mobility patterns, as well as a better understanding of the “pulse” of a particular city.

5 Conclusion

This research focused on investigating the dynamic mobility patterns of urban areas. We demonstrated that DTW is a highly effective method for exploring the similarity / dissimilarity of urban mobility patterns. The results indicate that the study area has the highest mobility density around 9am and 6pm, and this pattern exists for both weekdays and weekends. We also looked into the internal structures of the abnormal series. In addition, we provided a method to examine the similarity between a benchmark series and study areas based on the DTW distance matrix. The outlier detection method discussed in Section 4.2 can also be used to identify abnormal mobility patterns in future urban studies, as well as providing reference for transportation and urban planning.

This research provides us with new insights for modeling the changing mobility patterns for urban areas. Here we used Voronoi polygons to divide the study area, in future studies we will use grid cells (500m*500m) and compare both results. Moreover, in this paper the data is segmented into 1 hour granularity. It would be interesting to investigate how different temporal granularities impact the results. Age and gender factors of phone users should also be included in further studies. Another potential direction for future research is to investigate how the predefined local and global constraints affect the DTW distance and classification results. The methodology discussed in this paper can be applied to other cities. Moreover, DTW can also be used to examine individual mobility patterns of phone users (i.e., characterizing user trajectories based on the abnormality of visited areas).

References

1. Hägerstrand, T.: What About People in Regional Science? Papers of the Regional Science Association 24, 7–21 (1970)
2. Batty, M.: Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals. MIT Press, Cambridge (2005)
3. Harvey, A.S., Taylor, M.E.: Activity Settings and Travel Behaviour: A Social Contact Perspective. *Transportation* 27, 53–73 (2000)
4. Ratti, C., Sevtsuk, A., Huang, S., Pailer, R.: Mobile Landscapes: Graz in Real Time. In: The 3rd Symposium on LBS & TeleCartography, Vienna, Austria (2005)
5. Miller, H.: Geographic Data Mining and Knowledge Discovery: An Overview. In: Miller, H.J., Han, J. (eds.) *Geographic Data Mining and Knowledge Discovery*, 2nd edn., pp. 3–32. CRC Press, London (2009)
6. Hamilton, B.W.: Wasteful Commuting. *Journal of Political Economy* 90, 1035–1053 (1982)
7. Gordon, P., Kumar, A., Richardson, H.W.: The Influence of Metropolitan Spatial Structure on Commuting Time. *Journal of Urban Economics* 26, 138–151 (1989)
8. Phithakitnukoon, S., Horanont, T., Di Lorenzo, G., Shibasaki, R., Ratti, C.: Activity-Aware Map: Identifying Human Daily Activity Pattern Using Mobile Phone Data. In: Salah, A.A., Gevers, T., Sebe, N., Vinciarelli, A. (eds.) *HBU 2010. LNCS*, vol. 6219, pp. 14–25. Springer, Heidelberg (2010)
9. da Costa Filho, A.C.B., de Brito Filho, J.P., de Araujo, R.E., Benevides, C.A.: Infrared-Based System for Vehicle Classification. In: 2009 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC), pp. 537–540 (2009)
10. Larsen, J., Urry, J., Axhausen, K.W.: *Mobilities, Networks, Geographies*. Ashgate, Aldershot (2006)
11. Ahas, R., Mark, Ü.: Location Services - New Challenges for Planning and Public Administration? *Futures* 37, 547–561 (2005)
12. Yuan, Y., Raubal, M., Liu, Y.: Correlating Mobile Phone Usage and Travel Behavior - a Case Study of Harbin, China. *Computers, Environment and Urban Systems* 36, 118–130 (2012)
13. Gonzalez, M.C., Hidalgo, C.A., Barabasi, A.L.: Understanding Individual Human Mobility Patterns. *Nature* 453, 779–782 (2008)
14. Song, C.M., Qu, Z.H., Blumm, N., Barabasi, A.L.: Limits of Predictability in Human Mobility. *Science* 327, 1018–1021 (2010)
15. Ahas, R., Aasa, A., Mark, U., Pae, T., Kull, A.: Seasonal Tourism Spaces in Estonia: Case Study with Mobile Positioning Data. *Tourism Management* 28, 898–910 (2007)
16. Schwanen, T., Kwan, M.P.: The Internet, Mobile Phone and Space-Time Constraints. *Geoforum* 39, 1362–1377 (2008)
17. Couclelis, H.: Pizza over the Internet: E-Commerce, the Fragmentation of Activity and the Tyranny of the Region. *Entrepreneurship and Regional Development* 16, 41–54 (2004)
18. Kang, C., Ma, X., Tong, D., Liu, Y.: Intra-Urban Human Mobility Patterns: An Urban Morphology Perspective. *Physica A: Statistical Mechanics and its Applications* 391, 1702–1717 (2012)
19. Senin, P.: Dynamic Time Warping Algorithm Review. University of Hawaii at Manoa (2008)
20. Gunopulos, D., Das, G.: Time Series Similarity Measures and Time Series Indexing. *Sigmod Record* 30, 624–624 (2001)

21. Eiter, T., Mannila, H.: Computing Discrete Fréchet Distance. Christian Doppler Laboratory for Expert Systems (1994)
22. Ahn, H.-K., Knauer, C., Scherfenberg, M., Schlipf, L., Vigneron, A.: Computing the Discrete Fréchet Distance with Imprecise Input. In: Cheong, O., Chwa, K.-Y., Park, K. (eds.) ISAAC 2010, Part II. LNCS, vol. 6507, pp. 422–433. Springer, Heidelberg (2010)
23. Sakoe, H., Chiba, S.: Dynamic-Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Transactions on Acoustics Speech and Signal Processing* 26, 43–49 (1978)
24. Brown, M.K., Rabiner, L.R.: Dynamic Time Warping for Isolated Word Recognition Based on Ordered Graph Searching Techniques. In: International Conference on Acoustics, Speech, and Signal Processing, pp. 1255–1258 (1982)
25. Lee, J.-G., Han, J., Li, X., Gonzalez, H.: Traclass: Trajectory Classification Using Hierarchical Region-Based and Trajectory-Based Clustering. In: International Conference on Very Large Data Base (VLDB 2008), Auckland, New Zealand (2008)
26. Brown, J.C., Hodgins-Davis, A., Miller, P.J.O.: Classification of Vocalizations of Killer Whales Using Dynamic Time Warping. *Journal of the Acoustical Society of America* 119, E134–E140 (2006)
27. Krauß, T., Reinartz, P., Lehner, M., Schroeder, M., Stilla, U.: Dem Generation from Very High Resolution Stereo Satellite Data in Urban Areas Using Dynamic Programming. In: International Archives of the Photogrammetry. Remote Sensing and Spatial Information Sciences, vol. 36. Hannover (2005)
28. Nguyen, K.A., Zhang, H., Stewart, R.A.: Application of Dynamic Time Warping Algorithm in Prototype Selection for the Disaggregation of Domestic Water Flow Data into End Use Events. In: 34th IAHR World Congress, Brisbane, Australia, pp. 2137–2144 (2011)
29. Zhu, T., Wang, J., Lv, W.: Outlier Mining Based Automatic Incident Detection on Urban Arterial Road. In: The 6th International Conference on Mobile Technology, Application & Systems (Mobility 2009). ACM, Nice (2009)
30. Mardia, K.V., Kent, J.T., Bibby, J.M.: *Multivariate Analysis*. Academic Press, London (1979)

Author Index

- Abdalla, Amin 1
Abdelmoty, Alia I. 340
Anderson-Tarver, Chris 15
- Beard, Kate 160
Bell, Scott 258
Buchin, Kevin 29
Buchin, Maike 43
Buttenfield, Barbara 15
- Chiang, Yao-Yi 57
Chopard, Bastien 116
Ciferri, Cristina Dutra de Aguiar 173
Ciferri, Ricardo Rodrigues 173
- Dodge, Somayeh 43
Dorr, Christopher 146
Dube, Matthew P. 72
Dylla, Frank 212
- Egenhofer, Max J. 72
- Feng, Gefei 311
Frank, Andrew U. 1
- Gaffuri, Julien 87
Gleason, Mike 15
Gunturi, Venkata M.V. 325
- Janowicz, Krzysztof 102
Jones, Christopher B. 340
- Kauppinen, Tomi 102
Keßler, Carsten 102
Klippel, Alexander 212
Knoblock, Craig A. 57
- Li, Rongrong 187
Lin, Hui 187
- Maisonneuve, Nicolas 116
Mohan, Pradeep 132
- Nittel, Silvia 146
- Raubal, Martin 354
Rude, Avinash 160
- Shekhar, Shashi 132, 325
Siqueira, Thiago Luís Lopes 173
Speckmann, Bettina 29, 43
Stanislawski, Larry 15
Sun, Guibo 187
- Tanasescu, Vlad 340
Times, Valéria Cesário 173
Touya, Guillaume 198
- Verdonschot, Sander 29
- Wallgrün, Jan Oliver 212
Walton, Lisa A. 226
Watanabe, Toyohide 241
Wei, Ting 258
Whittier, John C. 146
Wiegand, Nancy 270, 284
Worboys, Michael 226, 298
- Yamamoto, Kosuke 241
Yang, Hui 311
Yang, Jinlong 212
Yang, KwangSoo 325
Younis, Eman M.G. 340
Yuan, Yihong 354
- Zhou, Xun 132