Configraphics
Graph Theoretical Methods for Design and Analysis of Spatial Configurations

Nourian, Pirouz

DOI
10.7480/abe.2016.14

Publication date
2016

Document Version
Final published version

Citation (APA)
Nourian, P. (2016). *Configraphics: Graph Theoretical Methods for Design and Analysis of Spatial Configurations*. [Dissertation (TU Delft), Delft University of Technology]. A+BE | Architecture and the Built Environment. https://doi.org/10.7480/abe.2016.14

A+BE
Architecture and the Built environment
#14
2016

Configraphics

Graph Theoretical Methods for Design and Analysis of Spatial Configurations

Pirouz Nourian

Configraphics

Graph Theoretical Methods for Design and Analysis of Spatial Configurations

Pirouz Nourian
Delft University of Technology, Faculty of Architecture and the Built Environment,
Department of Architectural Engineering + Technology / Department of Urbanism

Architecture and the Built environment

abe.tudelft.nl

Design: Sirene Ontwerpers, Rotterdam
Cover Image: Designed & Illustrated by Tim Hartin of Paratime Design in 2013, www.paratime.ca, courtesy of Tim Hartin, not available for commercial reproduction.

ISBN 978-94-6186-720-9
ISSN 2212-3202

Configraphics

Graph Theoretical Methods for Design and Analysis of Spatial Configurations

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op vrijdag 30 september 2016 om 12:30 uur
door Pirouz NOURIAN GHADI KOLAEE
Master of Science in Architecture, Tehran University of Art
Bachelor of Science in Control Engineering, KNTU, Tehran
geboren te Teheran, IRAN

Dit proefschrift is goedgekeurd door de

Promotor: Prof. Dr.Ir. I.S.Sariyildiz, Design Informatics
Copromotor: Dr.Ir. F.D. van der Hoeven, Urban Design

Samenstelling promotiecommissie bestaat uit

Rector Magnificus, Voorzitter
Prof. Dr.Ir. I.S.Sariyildiz, Technische Universiteit Delft, promotor
Dr.Ir. F.D. van der Hoeven, Technische Universiteit Delft, copromotor

Onafhankelijke leden

Prof. M.Batty, University College London, United Kingdom
Prof. I. Horvath Technische Universiteit Delft, the Netherlands
Prof. B. de Vries, Technische Universiteit Eindhoven, the Netherlands
Dr. P. Vakili, Boston University, United States of America
Prof. P. Russel, Technische Universiteit Delft, the Netherlands
Dr. M. Turrin, Technische Universiteit Delft, the Netherlands

Configraphics

Graph Theoretical Methods for Design and Analysis of Spatial Configurations

Dissertation

for obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus Prof. Ir. K. C.A.M. Luyben;
Chair of the Board for Doctorates
to be defended publicly on Friday, September 30, 2016 at 12:30 hours
by Pirouz NOURIAN GHADI KOLAEE
Master of Science in Architecture, Tehran University of Art
Bachelor of Science in Control Engineering, KNTU, Tehran
born in Tehran, IRAN

This dissertation is approved by

Promoter: Prof. Dr.Ir. I.S.Sariyildiz, Design Informatics
Copromoter: Dr.Ir. F.D. van der Hoeven, Urban Design

Composition of the Doctoral Committee

Rector Magnificus, Chairman
Prof. Dr.Ir. I.S.Sariyildiz, Technische Universiteit Delft, promoter
Dr.Ir. F.D. van der Hoeven, Technische Universiteit Delft, copromoter

Independent Members

Prof. M.Batty, University College London, United Kingdom
Prof. I. Horvath Technische Universiteit Delft, the Netherlands
Prof. B. de Vries, Technische Universiteit Eindhoven, the Netherlands
Dr. P. Vakili, Boston University, United States of America
Prof. P. Russel, Technische Universiteit Delft, the Netherlands
Dr. M. Turrin, Technische Universiteit Delft, the Netherlands

Foreword

I started this research with a dream to make a systematic design process based on spatial configuration. The core of my idea was that architecture should be less preoccupied with form and be more focused on the configuration of spaces and its effect on the functioning of a building. Later I went further with this idea to urban scale and theoretically studied the effect of spatial configuration on accessibility and mobility potentials. My fascination with Graph Theory and its potential applications in architectural design and analysis of built environment was another motive behind this work.

I began to develop interest in computational design methods, as a teaching assistant, back in 2006: I wanted to have some solid basis for my design teachings, being able to suggest 'methods' for designing buildings in a rational manner. That was how the ideas behind this work came to existence.

Now that I have finished this research, I cannot claim to have found perfect answers for all initial questions; but I hope to have exemplified the possibility of systematically approaching design through configuration. I hope the results of this thesis will turn useful or inspiring for projects that can potentially make meaningful differences in the lives of people and the planet, e.g. by facilitating planning processes in favour of cycling and walking for urban transportation or designing well-functioning buildings improving safety, security and working efficiency of their inhabitants.

Acknowledgements

I would like to thank many people who helped me in various ways in pursuing my interests and conducting this complex research. Apart from the research, my life as a researcher would have been very difficult without the kind support and pleasant presence of these people whom I hope to have remembered all and properly acknowledged their support. I hereby apologize to those whose names I have forgotten to mention.

I am very grateful to my promoter Prof. Dr. Ir. Sevil Sariyildiz, for accepting me in her research group, providing constructive supervision, and supporting my professional development in best ways imaginable. She has been very patient and attentive in her supervision, while kindly letting me find my own way. Her advices and encouragements have helped me clarify my thoughts, strengthen my skills, and deepen my knowledge throughout these years. Moreover, she kindly trusted in my capabilities and gave me exceptional teaching opportunities, which were synergistically very helpful for my research. Sevil has a remarkably positive character that continues to be inspiring and enthusing.

I sincerely thank my co-promoter Dr. Ir. Frank van der Hoeven for all the time he dedicated to my project. He was an inspiring and visionary advisor, motivating many developments in this work, while providing essential guidance on keeping the whole dissertation structured. He has also helped me much beyond the scope of this work in my professional development. I learnt a lot from working with him on a research proposal for EU grants. Frank also provided me an opportunity to work as a researcher on Future Internet software applications for Smart Cities, which was very inspiring for my future research.

I would like to thank the external and internal jury members of my PhD defence committee, respectively: Prof. Michael Batty from UCL/Centre for Advanced Spatial Analysis, Prof. Imre Horvath from TU Delft/Industrial Design/Computer Aided Design Engineering, Prof. Bauke de Vries from TU Eindhoven/Design Systems Group, and Dr. Pirooz Vakili from Boston University/Center for Information & Systems Engineering, as well as Prof. Peter Russel, the dean of TU Delft Faculty of Architecture and the Built Environment and Dr. Michela Turrin, assistant professor of Design Informatics for the time they kindly dedicated to assess my work. I am very grateful for the extensive feedback reports and meetings offered by Prof. Horvath and Dr. Vakili. Their inputs were very helpful in improving the current book.

I am very grateful to the designer of the A+BE book series Véro Crickx for the exceptional and amazing work she did on the layout of this book.

I would like to extend my gratitude to the following academics:

- Dr. Ir. Rudi Stouffs. He admitted me as a PhD researcher and introduced me to Prof. Sariyildiz. Rudi was my supervisor in my first year (before going to National University of Singapore). He gave me very helpful directions, offered me invaluable teaching opportunities, and encouraged me in pursuing my research ideas.
- Dr. Michela Turrin. She helped me structure my initial research proposal back in 2010. Michela has been ever since a great colleague and a wise and caring adviser to consult with every so often.
- Dr. Meta Berghauser Pont from the department of Urbanism, chair of Design Theory and Methods. She was my daily supervisor in 2011 and 2012, prior to going to Chalmers University. Meta motivated me to develop my ideas on computational urban design and helped me formulate the conceptual framework of my research.
- Ir. Paul de Ruiter. He has kindly supported me in multiple teaching positions in the past six years as the education manager of our research group. Paul has always kindly supported my professional development.
- Dr. Sisi Zlatanova from the department of Urbanism, Chair of 3D Geoinformation. She was my supervisor in an invaluable part-time research project on Voxels and Voxelization Algorithms for Modelling Built Environments. Throughout this project, as well as our collaboration in teaching the course 3D Modelling of Built Environment within the MSc Geomatics programme, I have learned a lot from Sisi. Our collaboration has strengthened my research by improving my technical skills and sharpening certain technical concepts.
- Dr. Nimish Biloria from Hyperbody research group for his sincere friendship and his invaluable advices on design teaching.

I would like to thank Dr. Akkelies van Nes, Prof. Juval Portugali, and Prof. Machiel van Dorst, from the department of Urbanism, for the lessons I learnt from their classes in the EMU post-graduate programme. Prof. Portugali also provided essential feedback on my conceptual framework back in 2012.

I express thanks to our co-authors of a joint research proposal called Bipedalism (on promotion of walking and cycling through computational planning support tools), led by Dr. Ir. Frank van der Hoeven. My special thanks to Ir. Ciaran McKeon from Transport Insights Ltd, Dr. Goncalo Corriera from TU Delf/Civil Engineering/Transport Planning, Dr. Keyvan Karimi from UCL/the Bartlett/Space Group, especially for his advice on integration of navigation costs into my accessibility models, Ir. Atakan Guven from Space Syntax Ltd, and Dr. Jorge Gil for their commitment and valuable lessons I learnt from them.

TOC

I would like to thank my dear friends who have motivated and helped me in my work in various ways, namely:

- Dr. Jorge Gil. He has given very valuable directions and suggestions on my work. I have learnt a lot from Jorge throughout our extensive collaboration on proposal writing and many fruitful conversations.
- Dr. Jose Nuno Beirao. He was a great colleague who helped me quickly get on track and start producing research results. Our collaboration was very exciting and inspiring, a truly unique experience!
- Ir. Barida Mashhoodi. Bardia has been a loyal companion in difficult academic adventures.
- Ir. Dena Kasraian. Dena has been a compassionate comrade and a confidant.
- Barida and Dena were the wise and kind friends whom I have been consulting on a daily basis during the past six years about the scientific aspects of my work as well as personal matters. They helped me thrive in the faculty and my career.
- Ir. Richard Schaffranek. Richard is a humble genius from TU Wien whose acquaintance made a big difference in the course of my work. I have been inspired by his great work on the application of graph theory in architecture and urban design.

I would like to thank my exceptional (former) students, Ir. Rusne Sileryte, Ir. Martin Dennemark , and Ir. Bart van Walderveen who have inspired me and kindly helped me in developing my work.

I am very thankful to David Rutten from McNeel Company, the visionary inventor who provided the great laboratory tools I needed.

I would like to thank Bo Song, Danielle Karakuza, Francoise Puffelen, Barbara van Vliet, Michel Oldenhof, Shaida Freese, Annemieke Berger, Nathalie Bast, and Linda van Keeken from administrative staff of TU Delft for their kind support.

I would like to thank fellow lab-mates, colleagues, collaborators, and friends: Dr. Ahu Sokmenoglu, Henry Kiksen, Dr. Ken Arroyo Ahori, Ir. Arman Norouzian, Dr. Zhiyong Wang, Ir. Liu Liu, Dr. Giancarlo Mangone, Ir. Mauricio Morales Beltran, Dr. Romulo Goncalves, Dipl.Ing. Alex Wandl, Dr. Olgu Chaliskan, Dr. Ir. Radan Suba, Ir. Yannis Chatzikonstantinou, Verena Balz, Ir. Anh Vu Vo, Ir. Achilleas Psyllidis, Ir. Gary (Jia-Ray) Chang, Ir. Ierm Erbas, Ir. Thaleia Konstantinou, Dr. Tomasz Jaskiewicz, Ir. Michiel Fremouw, and Ir. Milou Teeling for the enjoyable moments we have had talking, working, or socializing together and what I have learnt from them.

I would like to commemorate my departed father Ata Nourian, whose legacy has been a great inspiration for intellectual hard work. I thank my family members especially my mother Forouzandeh Mahfooz, for devoting her life to my development and cultivating the love of nature, science, research, humanity and music by setting a good example by her life, my aunt Fahimeh Mahfooz for being an intellectual coach, my uncle Ir. Farid Mostafavi for sparking the love of mathematics and engineering in me, my brother Bamdad Nourian for inspiring the love of arts and design, Sina Mostafavi for being my very first friend, long-time collaborator, and motivator, and Sarah Mostafavi for being a constant source of hope and happiness. I extend my gratitude to my larger family and a circle of close friends who I hope will excuse me for not having mentioned them.

I am deeply grateful to my partner, frequent co-author, and editor in chief, Samaneh Rezvani for truly being my partner in life and her great contributions to this work.

Contents

Foreword 7
Acknowledgements 9
Summary 21
Samenvatting 23

1 Introduction 25

1.1 Background and Necessity 25

1.2 Synopsis 27

1.3 Research Context and Scope 28

1.4 Audience 29

1.5 Problem Statement 29

1.5.1 Design Problems 30
1.5.2 Research Problems 31

1.6 Research Goals 32

1.7 Research Questions 33

1.8 Research Scope and its Limits 33

1.9 Position within Related Research Fields 34

1.10 Research Methodology 36

1.10.1 Literature Review 39
1.10.2 Problem Formulation and Conceptual Design 40
1.10.3 Implementation and Test Cycles 41
1.10.4 Research Tools and Techniques 42
1.10.5 Assessment and Adaptation 42

1.11 Related Tools and Methods 43

1.12 Scientific and Societal Relevance 45

1.13 Technology Readiness Level 46

1.14 Outline of the Dissertation 46

2 Research Methodology 49

2.1 Introduction 49

2.2 Background & Definitions 50

2.2.1 What is special about design problems? 50
2.2.2 Problem Formulation vs Problem-Solving 52
2.2.3 On Automated Design 53
2.2.4 Logical Leap in Design 54
2.2.5 Design Methodology & Design Research 54

2.3 Theoretical Underpinnings 56

2.3.1 Design Epistemology: Design Paradigms 56
2.3.2 Design Praxeology: Design Process & Design Methods 58
2.3.3 Design Phenomenology: Design as Spatial Configuration 59
2.3.4 Causality, Limitations of Models and Decision Support 60
2.3.5 Analysis vs Evaluation 61

2.4 On Computational Design 62

2.4.1 Computer Aided Design (CAD) and Parametric CAD 62
2.4.2 Design Space Exploration 63
2.4.3 Real-Time Analysis and Geo-Design 63
2.4.4 Feedback vs Feedforward 64

2.5 On Spatial Analysis 64

2.5.1 What is special about spatial analysis? 65
2.5.2 What-If Scenarios 67

2.6 Highlights 68

2.7 Methodological Approach 68

2.7.1 Theoretical Reflection 69
2.7.2 Problem Formulation & Concept Development 70
2.7.3 Mathematical Modelling 70
2.7.4 Algorithm Design 71
2.7.5 Software Development 71
2.7.6 Verification and Validation 72
2.7.7 Crowd-Sourced Test & Validation 74

3 Model and Methodology A: Architectural Configuration 75

3.1 Background and Motivation 76

3.2 Advantages of a Configurative Approach to Design 77

3.3 Definition of Architectural Configuration 79

3.3.1 Bubble Diagrams 79
3.3.2 Mathematical Definition 81
3.3.3 REL charts and From-To Charts 81

3.4 Architectural Spatial Network Modelled as a Graph 87

3.5 Preliminaries of Modelling Spatial Networks 88

3.6 Configurative Design Methodology 92

3.7 Configurational Analysis vs Configurational Synthesis 95

3.8 Configurative Design Process 96

3.8.1 Reading a Configuration Graph 97
3.8.2 Drawing Bubble Diagrams 99
3.8.3 Drawing Justified Graphs 101
3.8.4 Real-Time Space Syntax Analysis 102

3.9 Spatial Way-Finding and Geodesics in Buildings 102

3.10 Analysing Architectural Configurations 104

3.10.1 Depth (Space Syntax variant of Graph Theoretical Distance) 106
3.10.2 Integration (Space Syntax variant of Closeness Centrality) 107
3.10.3 Difference Factor 107
3.10.4 Entropy (Space Syntax variant of Shannon Index of Information Entropy) 108
3.10.5 Control 109
3.10.6 Choice (Space Syntax variant of Betweenness Centrality) 110
3.10.7 Random Walk Value (probability of presence) 110
3.10.8 Analysis versus Evaluation of Spatial Performance 111

3.11 Synthesising Architectural Configurations 111

3.11.1 Producing a Convex Embedding of the Connectivity Graph 116
3.11.2 Maximal Planar Graphs and Triangulation 118
3.11.3 Finding Dual Spatial Layout Topologies 118
3.11.4 Finding All Possible Adjacencies as Maximal Planar Graphs 119
3.11.5 Topological Possibility versus Geometrical Possibility 125
3.11.6 A Note on Rectangular Floor Plan Layout 126
3.11.7 Radical Axis, Power Diagrams, and Alpha Complexes (Shapes) 127
3.11.8 Towards freeform geometric layouts 129

3.12 2D Isovist Bubbles 131

3.13 3D Isovist Bubbles 135

3.14 Discussion: a new way of designing buildings 137

3.15 Future Work 139

4 Implementation & Test A: SYNTACTIC 141

4.1 Introducing SYNTACTIC: a toolkit for architectural configuration 142

4.2 Goals, Outlook and Target Users 143

4.3 Designer-Computer Interface 145

4.4 Design Workflow 145

4.5 Tools 148

4.5.1 Graph Formation and Graph Drawing (Figure 53) 148
4.5.2 Space Syntax Analysis (Figure 54) 149
4.5.3 Topological Embedding for Plan Layout (Figure 55) 150
4.5.4 Geometric Plan Layout (Figure 56) 151
4.5.5 Isovist Bubbles for an Agent-Based Model for Spatial Layout (Figure 57 & Figure 58) 154
4.5.6 Urban Configuration Analysis Tools (Figure 59) 155

4.6 Educational Use 156

4.7 Achievements and Limitations 161

4.8 Future Work 162

5 Model and Methodology B: Urban Configuration 165

5.1 Motivation, Societal Relevance and Scientific Relevance 166

5.2 Research Background and Context 169

5.3 Definition of Urban Configuration 170

5.3.1 Space versus Place: a Theoretical Clarification 171
5.3.2 Network Structure, Density, Diversity 172
5.3.3 Reconstructing a Neighbourhood Mathematically 172
5.3.4 Matter of Scale in Urban Analysis: Global vs. Local 175
5.3.5 A Chicken and Egg Problem 175

5.4 Urban Spatial Network Modelled as a Configuration Graph 176

5.4.1 Primal Undirected Graph, Undirected Network 183
5.4.2 Dual Directed Graph, Doubly-Directed Network 184
5.4.3 Dual Directed Graph, Undirected Network 185
5.4.4 A Unifying Framework (Architectural, Urban, Spatial) 186

5.5 Way Finding and Geodesics in Urban Environments 187

5.5.1 Walking and Cycling Impedance in Built Environment 188
5.5.1.1 Physical Impedance: how long and how steep 189
5.5.1.2 Cognitive Impedance: how difficult to navigate 191
5.5.2 Easiest Path Algorithm 194
5.5.3 Distance Redefined 197

5.6 Fuzzy Interpretation of Distance Measures as Closeness 198

5.7 Fuzzy Accessibility Measures for Pedestrians and Cyclists 200

5.7.1 Proximity (closeness to all POI) 201
5.7.2 Vicinity (closeness to any POI) 201
5.7.3 Fuzzy framework of CONFIGRAPHIX 201

5.8 Catchment as Crisp Closeness 209

5.9 Zoning for Facility Location and Business Intelligence 209

5.10 On Network Centrality Models 212

5.11 Geodesic Centrality Models 213

5.11.1 Betweenness, Local Betweenness 213
5.11.2 Closeness, Local Closeness 217

5.12 Spectral Centrality Measures on Configuration Graphs 220

5.12.1 A Very Short Introduction to Spectral Graph Theory 224
5.12.2 Degree Centrality 227
5.12.3 Eigenvector Centrality Metrics (Katz, Gould, Bonacich) 228

5.13 Probabilistic Models on Configuration Graphs 236

5.13.1 A Markov Chain Model/ Random Walks on Streets 237
5.13.2 Four Different Random Walk models 239
5.13.3 Stationary Distributions of Undirected Graphs 247
5.13.4 Stationary Distributions of Directed Graphs 257
5.13.5 Culprit On the Run and the Souvenir Shop 262

5.14 Future Work (Limitations and Open Problems) 266

5.15 Conclusion (Summary of Achievements) 268

6 Implementation & Test B: CONFIGURBANIST 271

6.1 Introducing CONFIGURBANIST: a toolkit for urban configuration analysis 272

6.2 Goals, Outlook and Target Users 273

6.3 Designer-Computer Interface 274

6.4 The Urban Configuration Analysis Workflow 275

6.4.1 Directed Graph from Doubly Directed Streets 275
6.4.2 Directed Graph from Undirected Streets 280

6.5 Tools 283

6.5.1 Map Simplification and Generalization Tools (Figure 146) 283
6.5.2 Topological Modelling Tools (Figure 147) 284
6.5.3 Graph Construction and Graph Traversal Tools (Figure 148) 285
6.5.4 Polycentric Accessibility Measures (Figure 149) 286
6.5.5 Inputs for Polycentric Accessibility Analysis Tools (Figure 150) 287
6.5.6 Geodesic Centrality Measures (Figure 151) 288
6.5.7 Spectral Centrality Measures (Figure 152) 289
6.5.8 Random Walk Probabilistic Models (Figure 153) 290
6.5.9 Spectral Graph Drawing Tools (Figure 154) 291
6.5.10 Matrix Plot Tools (Figure 155) 292
6.5.11 Graph Drawing Utility Tools (Figure 156) 293
6.5.12 Quantitative Validation & Calibration Tools (Figure 157) 294

6.6 Implementation Details and Issues 296

6.6.1 Network Topological Models 296
6.6.1.1 Vector-Based Approach 296
6.6.1.2 Raster-Based Approach 300
6.6.2 Graph Data Models 301
6.6.3 Graph Search Algorithms 302
6.6.4 Linear Algebraic Algorithms 302
6.6.4.1 Eigenvector Centrality 303
6.6.4.2 Katz Centrality: 306
6.6.4.3 Few Dominant Eigenvectors 308

6.7 The Architecture of CONFIGRAPHICS.DLL 309

6.8 Generating Accessibility Evaluation Reports 314

6.9 Qualitative Evaluation 317

6.10 Future Work 318

7 Conclusions 319

7.1 Summary of Results 319

7.1.1 Results of Chapter 1 319
7.1.2 Results of Chapter 2 320
7.1.3 Results of Chapter 3 320
7.1.4 Results of Chapter 4 321
7.1.5 Results of Chapter 5 321
7.1.6 Results of Chapter 6 321

7.2 Response to Research Questions 322

7.3 Discussion 323

7.4 Future Work 325

7.4.1 Configurative Architectural Layout (Chapters 3, 4) 325
7.4.2 Spatial Configuration Analysis (Chapters 5, 6) 326

Endnotes 327
Works Cited 339
Curriculum Vitae 343

Summary

This dissertation reports a PhD research on mathematical-computational models, methods, and techniques for analysis, synthesis, and evaluation of spatial configurations in architecture and urban design. Spatial configuration is a technical term that refers to the particular way in which a set of spaces are connected to one another as a network. Spatial configuration affects safety, security, and efficiency of functioning of complex buildings by facilitating certain patterns of movement and/or impeding other patterns. In cities and suburban built environments, spatial configuration affects accessibilities and influences travel behavioural patterns, e.g. choosing walking and cycling for short trips instead of travelling by cars. As such, spatial configuration effectively influences the social, economic, and environmental functioning of cities and complex buildings, by conducting human movement patterns. In this research, graph theory is used to mathematically model spatial configurations in order to provide intuitive ways of studying and designing spatial arrangements for architects and urban designers. The methods and tools presented in this dissertation are applicable in:

- arranging spatial layouts based on configuration graphs, e.g. by using bubble diagrams to ensure certain spatial requirements and qualities in complex buildings; and
- analysing the potential effects of decisions on the likely spatial performance of buildings and on mobility patterns in built environments for systematic comparison of designs or plans, e.g. as to their aptitude for pedestrians and cyclists.

The dissertation reports two parallel tracks of work on architectural and urban configurations. The core concept of the architectural configuration track is the 'bubble diagram' and the core concept of the urban configuration track is the 'easiest paths' for walking and cycling. Walking and cycling have been chosen as the foci of this theme as they involve active physical, cognitive, and social encounter of people with built environments, all of which are influenced by spatial configuration. The methodologies presented in this dissertation have been implemented in design toolkits and made publicly available as freeware applications.

Keywords: Spatial Configuration, Architecture, Urban Design, Graph Theory, Mathematical Modelling, Computational Design

Samenvatting

Dit proefschrift meldt een promotieonderzoek op wiskundige-computationele modellen, methoden en technieken voor analyse, synthese en evaluatie van ruimtelijke configuraties in de architectuur en stedenbouw. Ruimtelijke configuratie is een technische term die verwijst naar de specifieke wijze waarop een aantal ruimten met elkaar verbonden als een netwerk. Ruimtelijke configuratie invloed op de veiligheid, beveiliging en efficiency van het functioneren van complexe gebouwen door het faciliteren van bepaalde patronen van beweging en / of belemmeren andere patronen. In steden en voorsteden bebouwde omgeving, ruimtelijke configuratie beïnvloedt bereikbaarheids en invloeden reizen gedragspatronen, bijv. de keuze van wandelen en fietsen voor korte reizen in plaats van reizen met de auto. Als zodanig, ruimtelijke configuratie daadwerkelijk invloed heeft op de sociale, economische en ecologische functioneren van steden en complexe gebouwen, door het uitvoeren van menselijke bewegingspatronen. In dit onderzoek wordt grafentheorie gebruikt om ruimtelijke configuraties te modelleren mathematisch om intuïtieve manieren van studeren en het ontwerpen van ruimtelijke regelingen voor architecten en stedenbouwkundigen te bieden. De methoden en instrumenten die in dit proefschrift zijn om ontwerpers te helpen bij:

- het regelen van de ruimtelijke lay-outs op basis van de configuratie van grafieken, bijv. door gebruik te maken bubbel diagrammen aan bepaalde ruimtelijke eisen en kwaliteiten in complexe gebouwen te waarborgen; en
- het analyseren van de mogelijke gevolgen van hun beslissingen over de te verwachten ruimtelijke prestaties van gebouwen en op mobiliteit patronen in de gebouwde omgeving, zodat ze ontwerpen kunnen vergelijken of plannen systematisch, bijv. met betrekking tot hun geschiktheid voor voetgangers en fietsers.

Het proefschrift rapporteert twee parallelle sporen van het werk op architectonische en stedenbouwkundige configuraties. Het kernbegrip van de architectonische configuratie spoor is de ‘bubbel diagram’ en het kernbegrip van het stedelijk configuratie spoor is de ‘makkelijkste paden’ voor wandelen en fietsen. Wandelen en fietsen zijn gekozen als de brandpunten van dit thema als ze te betrekken actieve fysieke, cognitieve en sociale ontmoeting van mensen met een gebouwde omgeving, die allemaal beïnvloed door de ruimtelijke configuratie. De methoden die in dit proefschrift zijn in design toolkits geïmplementeerd en het publiek beschikbaar als freeware gesteld.

1 Introduction

This chapter gives an overview of the research, the motivation behind it, the methods used and the structure of the research in relation to the research questions and problem formulations.

§ 1.1 Background and Necessity

In designing functionally complex buildings, e.g. hospitals, airports, etc., configurational requirements are very severe, in that the spatial configuration has an evident effect on the safety, security, and efficiency of functioning of such buildings. In larger built environments, i.e. in cities, spatial configuration evidently affects travel behaviour and accessibilities, which in turn affect the social status of neighbourhoods.

The social aspects of configuration are not only of interest in large and complex built environments, but also in smaller scales. There is evidence that many vernacular buildings form categories of configurations, i.e. their resemblance to each other is not merely formal but more deeply configurational ((Hillier, B., Hanson, J., 1984)& (Habraken, 1988)) and that the spatial configuration both reflects and affects patterns of social interaction (ibid.). It is therefore important to know how exactly configuration affects the social functioning of buildings. Interestingly, spatial configurations are ‘understandable’ for both humans and computer programs once modelled as graphs (networks); therefore, a configurative design process can be intuitive and structured at the same time. We are interested to see if architectural design can be structured as a process of going from abstract configurational ideas to concrete geometric designs, i.e. by focusing on the most fundamental aspect of architecture that is the spatial structure[1].

Nobel laureate Herbert A. Simon wrote in his famous book The Sciences of the Artificial (Simon, 1999, pp. 151-152) argued about a problem for architects, that still seems to hold true:

“... [An] increasingly acute problem for architects is that, when they take on the task of designing whole complexes or areas instead of single buildings, their professional training does not provide them with clear design criteria. In city planning, for example, the boundary between the design of physical structures and the design of social systems dissolves almost completely. Since there is little in the knowledge base or portfolio of

TOC

techniques of architecture that qualifies the professional to plan such social systems, the approach to the design tends to be highly idiosyncratic, reflecting little that can be described as professional consensus, and even less that can be described as empirically based analytic technique."

It was exactly because of this deficiency in the architecture curriculum that we thought of this research in the first place.

Configurational analysis in urban studies is fundamentally important. Without considering configuration, spatial analyses might fail in capturing the wholeness of urban phenomena in their spatial manifestation. An apparent example can be studying land-use mix or population density in a 2D raster tiled with square miles. Having geographic coordinates might seem quite adequate; but Euclidean coordinates per se suggest that straight lines would indicate the distance between locations, an assumption that is mostly false in real urban environments. It is obvious that if two blocks are on the opposite sides of a river or a highway, their access to one another are not through their straight line of sight but through a possibly much longer route. It is therefore suggested that a discrete network based spatial representation is more suited for urban studies, especially those involving human actions, as it facilitates consideration of actual network distance. Furthermore, topological relations can be used in structuring big urban data and in making 'spatial sense' of them. For instance, we can measure density and diversity along the network instead of looking at tiles (pixels) of a raster. Such an analysis would correspond much better to what is experienced by people in urban space (e.g. streets), compared to density or diversity per square kilometre. In short, we can say that looking at urban settings without considering connections is like looking at parts without seeing the whole.

The general objective of this research is to contribute primarily to the process of construction of the foundations of an [emerging] science of design and planning that focuses on the spatial structures and their effect on such things as mobility, accessibility, and social interactions. We advocate an evidence-based or performance-driven design; in which, normative arguments about good practice in design are reconsidered by seeking theories that could explain the actual measured behaviour of people. The advent of new information and data gathering technologies could facilitate validation and calibration of models and methods used in this approach and the landscape of architecture and built environment research can change in favour of forming a body of knowledge that we can consider as design sciences. Our main contribution to this approach could be described as inventing methods and sharpening the tools of measurement for spatial analysis.

§ 1.2 Synopsis

In this dissertation, we will introduce a graph theoretical methodology for architectural design and urban configuration analysis, for studying walking and cycling accessibility. Despite the seeming disparity between the two subjects, we will see that the two matters are closely related in the way they are being treated mathematically. For this reason, they are presented as combined in one conceptual framework that we call CONFIGRAPHICS as short form for 'graphics' (pertained to graph theory as in graphical models in probabilities) and 'configuration' analysis and synthesis. Configuration here refers to the spatial arrangement patterns that will be mathematically encoded in labelled graphs (networks), whose nodes are spatial units (e.g. rooms in buildings and streets in cities). The configuration graphs can also be used to represent the spatial arrangement of such things as density (population, built space) and diversity (land-use mix). In a sense, the idea of representing buildings and urban configurations as graphs (a.k.a. networks) sounds very uncomplicated and straightforward, this is exactly why we focused on graphs in the first place: as representations that are equally understandable for humans and computer programs. We will see how addressing spatial configurations from a graph theoretical point of view can help in constructing knowledge about the functioning of buildings and cities and how it might help in approaching design in a systematic manner. By systematic we mean a process that is based on a method and clear criteria for evaluation, but certainly NOT an automated process.

In the case of architectural design, we will see if it is possible to design buildings starting from bubble diagrams. By representing an architectural configuration as a bubble diagram, we can immediately analyse it in terms of its likely performance regarding the extent to which a spatial configuration fosters desired mobility potentials and provides for desirable social interactions in accordance to a functional programme[2]. The importance of this matter is best understood in the context of designing spatially and functionally complex buildings such as hospitals, museums, airports, etc.

A bubble diagram is considered as a labelled graph, which can give rise to many [but not too many] layout patterns and many geometric designs. If there are no geometric constraints, then the possibilities will be infinite; but with some constraints, we can see that the universe of possibilities might be finite and enumerable. We will introduce this methodology and discuss its potentials as a theoretical investigation into architectural design in terms of expansion of 'design space'. In addition, we will present the foundations of a novel computational geometry object for representing spatial nodes that provides for intuitive design of spatial configuration, i.e. configurative design.

The second theme is urban spatial configuration and its effect on 'active' mobility potentials and accessibility, i.e. by means of walking and cycling. It is intuitively understandable that the shape of built environment somehow affects mobility of people; the question is 'how exactly'? We will propose a methodology for analysing such effects using graph theory, linear algebra, and fuzzy logics. We will see how 'actual distance' is different from simplistic distance metrics and how an accurate temporal metric for distance could found a new basis for [modal] urban network analysis.
We propose a spatial network analysis methodology based on a novel optimal path algorithm that we call Easiest Path. This methodology allows for measuring relative closeness of locations to some or all possible destinations of interest (alias attraction points). Using the same spatial network representation used for accessibility analysis, we will present a family of random walk probabilistic models of passage of pedestrians and cyclists using mathematical constructs such as Markov Chains.

We first explain our research methodology in chapter 2. After introducing each of the devised computational methodologies (in chapters 3 and 5), we discuss their implantations (in chapters 4 and 6). We conclude this dissertation by reflecting on the initial research questions, summarizing the achievements and limitations, and identifying areas of necessary future work.

§ 1.3 Research Context and Scope

The research reported here is generally in the field of design computation and computational performance assessment, aimed at developing methods for supporting design and spatial decision-making in architecture and urban design. The research has two deliverables:

- a topological design methodology for architectural layout, and
- a Spatial Network Analysis library that can be used for assessing walking and cycling accessibility

The research addresses the areas of architectural morphology (Steadman, 1983), urban morphology (Moudon, 1997), architecture, urban planning and urban design, geo-computation, geo-design, and network studies. The tools of this research are mathematical formalisms and methods from Graph Theory, Topology, Linear Algebra, Combinatorics, Statistics and Probabilities, and Fuzzy Logics.

§ 1.4 Audience

This thesis is mainly targeted at researchers working in the areas of computational design and spatial analysis, particularly those interested in graph-theoretical approaches. Spatial Network Analysis is inspired by pioneering works in the area of Social Network Analysis; and as such, this work could be potentially of interest in that area as well. The area of work reported in this dissertation is inherently multidisciplinary, with a variety of topics ranging from computational network analysis, computational topology and computational geometry, spectral graph theory, and stochastic modelling.

The contents are essentially of mathematical and/or computational nature; however, the intended application areas, and the relevance of topics, are architectural or related to urban-design and planning. Therefore, we can envisage potentially interested audience from the whole range of these research fields. Considering this variety, and for the sake of brevity, we have tried to explain the mathematical and computational processes in a relatively plain language, avoiding proofs and formal definitions.

The novelties reported in this dissertation (thereby potential areas of interest) are of two types:

- Combining mathematical and/or computational methods in novel ways in order to apply them to new areas of application in architecture and urban design; or
- Designing new algorithms, mathematical models, and computational constructs

§ 1.5 Problem Statement

Architectural design practice is often preoccupied with shape and formal issues while 'the hidden structure of the space' (Hillier, 2007) is mostly neglected, such that the consequences of spatial decisions show up after the building is realized. Examples include doors, corridors and staircases that are exposed and inviting to all visitors but always have to be kept closed in public buildings. Other examples include cafés, shops, or supposedly public spaces that never gain popularity in public buildings, empty office buildings, and of course, the failed so-called city centres in new towns, 'crime havens' caused in some massive urban development projects and alike. The most sensible of all problems clearly pertained to spatial configuration can be recognized as those of 'circulation and access problems in complex buildings', 'poor accessibility in cities', 'car dependency', and 'urban sprawl' that cause severe economic and environmental problems. Note that transport sector has been responsible for up to 63.8 percent of oil consumption in the world (IEA (International Energy Agency), 2013, p. 33)[3].

In the following sections, we discuss the problems in their specific contexts, from a holistic point of view to technical problems pertained to the difficulties in addressing the main problems. We begin by sharpening the descriptions given above.

§ 1.5.1 Design Problems

Below, the key problems addressed in architectural and urban design are paired with corresponding propositions:

Problem: In functionally complex buildings such as museums, hospitals, airports, and alike, configurational problems can lead to ineffective circulation, costly maintenance, economical malfunction, or even dangerous operational problems such as those affecting safety or causing security leaks. Thus, configurational thinking needs to be at the core of design process to consider who should have what kind of access to what spaces and what spatial accesses need to be provided (or facilitated) or otherwise blocked (or impeded). There are no comprehensive approaches for addressing configurational design explicitly in architectural layout.

Proposition: We argue that we need an explicitly configurational approach for integrated design and analysis in architectural layout that is systematic, generic, and intuitive at the same time.

Problem: Spatial configuration in cities has a direct effect on walking and cycling potentials and accessibility; which in turn have direct impact on viability of many retail businesses, social integration, public health, social safety and security (social segregation and crime havens), and environment (car dependency and its consequences). The existing methods for spatial network analysis such as those of Space Syntax and alike or those of Transport Planning do not adequately address walking and cycling in their physical and cognitive entirety, especially in modelling paths, distances and travel-times. The network models used in these approaches have inherent shortcomings in addressing wayfinding in walking and cycling. Many earlier models, measures, and methods are too abstract or very difficult to interpret in terms of their real-world meaning.

Proposition: We argue that there is a need for a novel comprehensive approach to spatial network analysis in order to capture the physical and cognitive aspects of walking and cycling mobility and accessibility. The proposed approach should contain indicators that would be easily interpretable in terms of physical quantities (e.g. travel time).

§ 1.5.2 Research Problems

Considering the space that we are living in as a continuum, we can observe that our direct (i.e. straight-line) access is almost always spatially obstructed in buildings and cities; therefore the notion of Euclidean distance would be inadequate for modelling proximities and spatial distributions. Discrete models of space, which are composed of nodes representing 'units of space (e.g. convex spaces)' will be more practical than Euclidean representation of space in many cases of spatial analysis, especially in finding real-world spatial distances. Once accepting that a network representation of space is needed then we have to define a systematic way of representing spatial units as well as their connections. Any such approach would facilitate certain measurements and hinder some others. We have chosen to look at this matter from a design point of view. Therefore, we formulated the problem as finding ways through which spatial units and their connection can be modelled mathematically and computationally, in order to allow for a designer-friendly design process that is 'based on configuration'.

Different representations of urban spatial networks are generally distinguishable as street-to-street or junction-to-junction adjacency representations. The former is as old and established as graph theory itself and the latter is known in spatial analysis domains, such as Space Syntax. It can be generally said that the street-to-street representation is more powerful in dealing with cognitive aspects of way finding and therefore fitter for analysing active modes of transport (e.g. walking and cycling), which evidently have a lot to do with the perception of people from space and spatial configuration. This is exactly where an architectural viewpoint can be helpful in addressing these modes of mobility. Specifically in modelling walking and cycling, cognitive and physical ease of walking or cycling are obviously important when it comes to modelling people's preferences or choices in choosing these as their modes of transportation over other possibilities.

The existing configurational representations such as those of Space Syntax have shortcomings in dealing with particularities of places, spaces with geographic attributes, namely in addressing topographic streetscapes and actual distances. Besides, their indicators are mostly difficult to interpret physically. In other words, Space Syntax indicators of configuration qualities need to be interpreted by experts as to what they imply in real world and that these experts might insert their subjective view in the interpretations. Besides, there are a number of inconsistencies in the basic definitions of spatial units (e.g. axial lines[4]) and the definition of measures such as integration[5]. Furthermore, Space Syntax focuses on cognitive distance and disregards physical or travel-time distance. We aim at developing indicators that clearly refer to physical quantities such as probability of presence of people, closeness to some points of interest in the sense of temporal distance through Easiest Paths and alike. We aim to deliver alternative-complementary configurational models, methods, and measures that are physically interpretable and intuitively understandable at the same time.

§ 1.6 Research Goals

Considering the essential importance of spatial configuration in functioning of buildings, in accessibility of locations, and in mobility of people in cities, the general objective of this research is to propose computational methodologies for analysing and synthesizing architectural and urban configuration. These methodologies are intended to be intuitive, extendable, and easy to integrate with computational design workflows and spatial decision support systems. The constructs, models and methods are designed to be comprehendible for both professional and non-professionals in that they should only describe 'physically tangible entities' such as 'travel time', 'passage probabilities' in the case of accessibility models and applicable constructs such as 'a smart polygon that remains visible from a point and maintains its surface area'. The models and methods for urban configuration analysis should be adaptable to allow for taking account of context, i.e. qualities pertained to the geographical place. Thus, representing space in isolation from physical and geographical attributes (as in most Space Syntax methods) would not be sufficient. The specific goals are enlisted below:

- To deliver a design methodology for spatial layout that brings spatial network analysis to architectural design process for assessing social, functional, or programmatic performance of a building configuration;
- To deliver a methodology for analysing the effects of spatial configuration on walkability and bikeability; and
- To merge spatial analysis in architectural scale with urban scale network studies in a unified conceptual framework.

It is intended to avoid subjective accounts as much as possible in developing constructs and indicators. For instance, we avoid giving quantitative definition for such qualitative things as 'liveability' as much as possible. In addition, we focus on physical 'dimension' (or meaning) of all indicators developed. As mentioned above, we focus on the physical and cognitive aspects that are objectively measureable.

§ 1.7 Research Questions

This research is inherently design-oriented. This orientation to design is twofold: 1) the research is aimed at providing new workflows for spatial design and decision-making; and 2) the research is conducted to design and produce theories, models, methods, and constructs, NOT to test any hypotheses using existing models or theories. In other words, it does not seek explanations as to how things are; it instead seeks for new ways of making things. Therefore, the research has propositions instead of hypotheses and its questions take the form of methodical questions.

The main question of this research is formulated as below:

How can we model spatial performance in architecture and urban design?

- How can we obtain an architectural layout from a spatial configuration graph, while controlling its performance? (Chapters 2, 3, 4)
- How can we model the effect of spatial configuration on accessibility (e.g. by walking and cycling) and mobility potentials? (Chapters 2, 5, 6)
- How can we integrate architectural and urban spatial analyses and estimate the spatial performance of design proposals? (Chapters 6, 7)

§ 1.8 Research Scope and its Limits

Prior to developing mathematical and computational models, we have reflected on the nature of architectural and urban design processes, guided by an intensive study of the so-called design research discipline (Cross, 1999). Practically, however, the research requires design and development of computational models and methods for spatial network analysis and synthesis. The topics enlisted below are within the scope of this research:

- Computer Aided Architectural Design
- Graph Theoretical Modelling of Discrete Spatial Network Models
- Computational Topology, Geometry, and Graph Drawing
- Path-Finding Algorithms and Network Centrality Studies
- Analytic Kernel of] Design, Planning or Spatial Decision Support Systems (DSS, PSS, or SDSS respectively)
- Spectral Graph Theory and Stochastic Modelling using Markov Chains

The following topics are marginally related to this research but fall *outside its scope*:

- Building Information Modelling (BIM)
- Optimization of Architectural Layouts
- Optimization of Spatial Configurations
- Land Use Allocation and Density Distribution
- Travel Demand Modelling and Transportation Forecasting Models
- Land Use Transportation Interaction (LUTI) Models
- Continuous Spatial Models for Pedestrian Flow Modelling and Path Finding in Continuous Space
- Travel Behaviour Studies and Mode Choice Models for Slow Traffic
- Indoor/Outdoor Navigation, Wayfinding and Positioning Technologies
- Travel safety, Scenic or Sensory Pleasance of Urban Routes
- Validation of Volunteered Geographic Information (VGI) e.g. OpenStreetMap
- Human Computer Interaction in Design Praxeology
- Modelling and Simulation of Crowd Movements, Evacuation or Egress in Emergencies
- Schools of Thoughts in Design and Planning
- Philosophical Bases of Design Methodology
- Database Management Systems (DBMS) in Decision Support Systems
- Geographical Information Systems (GIS)

§ 1.9 Position within Related Research Fields

The specific reviews of state-of-the-art for each theme of this research are extensively given in pertinent chapters. Here we present the general trends and point to the underlying disciplinary structure of the fields dealing with subject matters of this research.

The research is predominantly carried out in two generic areas of research, namely **Computational Design** and **Spatial Analysis**.

In order to give an overview of the disciplinary status of our project, we introduce two key areas in this research, namely:

- architectural [and urban] design in a field that can be roughly called 'Computational Design', and
- configurational studies in the field of architectural and urban 'Spatial Analysis'.

Configurational modelling can be traced back to the early works of March and Steadman in UCL back in 70's (e.g. (March, L, Steadman, P, 1974)), and later to those of Hillier and Hanson in 80's (e.g. (Hillier, B., Hanson, J., 1984)). Late Alasdair Turner significantly contributed to the development of computational models and methods for spatial analysis. Michael Batty has contributed to the foundation of an interdisciplinary field that can be called 'Mathematical Modelling of Cities'. Researchers such as Bin Jiang are recognizable for their work in the field of Geo-Informatics in analysing urban spatial networks. Professional Computational Design practitioners such as Christian Derix (former head of Computational Design R&D group in AEDAS) have contributed to recognition of the field in practice. In Transportation Modelling research, few researchers such as Serge Hoogendoorn have worked on the foundation of mathematical models of pedestrian flows and mathematical modelling of the so-called slow traffic (walking and cycling).

For a long time (almost 25 years), Space Syntax was the only option available for studying spatial configurations. This situation has changed after the introduction of a few alternative methodologies. Studying large networks seemed to be something doable only via dedicated desktop GIS software applications; but that situation has also changed by introduction of web-based GIS applications and free/open-source geo-spatial DBMS such as PostGIS for PostgreSQL. Lastly, the availability of open geo data allows for processing networks in almost any environment capable of computation including CAD environments such as McNeel's Rhinoceros3D[a]. In short, performing urban configuration analyses is not limited to Space Syntax software (e.g. Depthmap) and GIS applications.

The use of conventional Spatial Decision Support Systems (SDSS) has been rather limited in design practice, for a variety of reasons (Uran, Oddrun, and Ron Janssen, 2003)[6], namely complicated user-interface, lack of capability for [plain] evaluation and ranking of scenarios, and most importantly lack of functionalities for easily creating design alternatives (ibid). The latter shortcoming could be best handled in a flow-based[b] computational design process, a paradigm in design technologies that is often referred to as parametric design (e.g. Grasshopper3D[c], Generative Components[d], Tygron[e], Flood[f]). The potentials of this new paradigm are among the motivations for choosing a parametric CAD environment as a testbed for implementation of the methods proposed in this research. Ultimately, however, the research is focused on the methods themselves, not on the technicality of implementing them in GIS or

a https://www.rhino3d.com/
b http://www.jpaulmorrison.com/fbp/
c http://www.grasshopper3d.com/
d https://www.bentley.com/en/products/product-line/modeling-and-visualization-software/generativecomponents
e http://www.tygron.com/
f https://www.floodeditor.com/

CAD environment. Therefore, choice of a host environment, i.e. the parametric CAD environment Grasshopper3D, for implementing and testing the methods is not central to the research, but rather a matter of convenience and practicality.

§ 1.10 Research Methodology

This is a "Research in design methodology", "Research in design technology" and "Research in design application" termed and explained in (Cross, 1999) and (Horvath, 2001). *This research utilizes scientific methods to devise models, methods, and techniques applied in architecture and urban design*. It has to be noted that this is NOT a behavioural science research. It is not our primary intention to propose theories on how people 'actually' move in buildings or in cities, at least not in the context of this dissertation. Instead, we will provide the methods that can help other researchers to study such phenomena. Therefore, the matter of validation of measures introduced in the research with empirical data falls outside the scope of this research[a].

To avoid further confusions regarding the difference between the **process** and the **products** of this research, we distinguish the meanings and differences of a few terms in the context of this thesis:

- Methodology: a structured collection of methods[b]
- Method: "a particular procedure for accomplishing or approaching something, especially a systematic or established one" (Oxford Dictionary)
- Technology: a structured collection of techniques[c]
- Technique: "a way of carrying out a particular task, especially the execution or performance of an artistic work or a scientific procedure"(ibid)
- Model: a mathematical/computational replica of a system, process, or construct

a It would be firstly impractical to expect a large body of validation studies comparable to those produced by Space Syntax community in a period of over 25 years; secondly, we argue that the matter of validation should be handled with more care, as we will suggest later in this dissertation. We will describe how the results of this research can be potentially validated later. More importantly, validation and verification would be best done by disinterested third parties.

b "A system of methods used in a particular area of study or activity" (Oxford Dictionary)

c A technology, literally meaning a science of craft, can be eventually operated by individuals that do not necessarily know deeply how to act following the underlying scientific methods. In that sense the important role of technology in this research is to bring scientific know how into practice.

In this work we reserve the terms models and methods for mathematical or algorithmic constructs and use the term technique for referring to constructs that pertain to programming languages, testbed environments, execution dependencies and alike.

To give an overview of how this research has been structured, we point to a workflow for **design science research**, which is a rather pragmatic way of structuring research methods as a methodology in the context of developing design or "spatial decision support systems" in the more general context of developing information or decision support systems. "Design Science Methodology" is described in two frequently cited papers on Design Science Research Methodology (March, Salvatore T., and Gerald F. Smith, 1995) that is specifically on 'designing Decision Support Systems' and another one that is on 'designing Information Systems in general' (Peffers, K, Tuunanen, T, Rothenburger, M A, Chatterjee, S, 2007). Software Engineering, Management, and Design Research (design methodology) have borrowed many terms, methods, and frameworks from each other in that they all deal with kinds of design activities that aim at developing new systems or changing systems for improving certain processes or situations.

"Historically and traditionally, it has been the task of the science disciplines to teach about natural things: how they are and how they work. It has been the task of engineering schools to teach about artificial things: how to make artifacts that have desired properties and how to design. Engineers are not the only professional designers. Everyone designs who devises courses of action aimed at changing existing situations into preferred ones. The intellectual activity that produces material artifacts is no different fundamentally from the one that prescribes remedies for a sick patient or the one that devises a new sales plan for a company or a social welfare policy for a state. Design, so construed, is the core of all professional training; it is the principal mark that distinguishes the professions from the sciences. Schools of engineering, as well as schools of architecture, business, education, law, and medicine, are all centrally concerned with the process of design" (Simon, 1999, p. 111) .

The 'design science' research process put forward by (Peffers, K, Tuunanen, T, Rothenburger, M A, Chatterjee, S, 2007) includes the following steps:

- Problem Identification and Motivation,
- Definition of Objectives for a Solution,
- Design and Development,
- Demonstration,
- Evaluation, and
- Communication

In other words, it can be seen as a process of:

- Conceptual Problem Formulation,
- Design and Development,
- Implementation,
- Verification, and
- Validations

March et al (March, Salvatore T., and Gerald F. Smith, 1995) define this process, in a slightly different manner, as:

- Build,
- Evaluate,
- Theorize, and
- Justify

In their definition, they identify the research products as:

- Constructs,
- Models,
- Methods, and
- Instantiations

This latter definition exactly describes the line of work reported in this dissertation. The mentioned steps have been followed iteratively through many cycles of conceptual development of mathematical models, design and implementation of algorithms, verification, crowd sourced test and validation, evaluation and theoretical reflection[7]. A summarized version of our research methodology is shown in Figure 1, in terms of the critical phases.

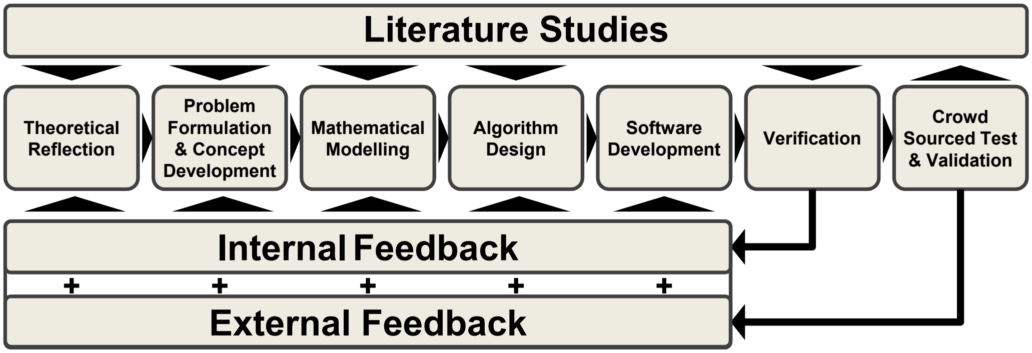


FIGURE 1 the research methodology of this dissertation summarized in terms of critical phases

This diagram is expanded in the next chapter, which goes into the details of the research methodology.

§ 1.10.1 Literature Review

There are different research communities sometimes working on similar issues that might not communicate to each other nor acknowledge each other's work unless there is wide public recognition of the work. Relevant examples of such diverse research communities are those of spatial analysis and in particular space syntax, geo-informatics, computer aided design and transportation planning. Being aware of this diversity and the differences in terminologies and jargons, we have sought traces of relevant research works in these areas. As this research is primarily architectural, the literature research was commenced with Space Syntax literature (initiated by Bill Hillier and Julienne Hanson)[a], extended then to the wider scope of Spatial Analysis research that is best presented by typical researches of the Centre for Advanced Spatial Analysis (CASA[b]) at University College London (led by Michael Batty). We have conducted a review of advanced computer aided [architectural] design CAD and CAAD methods[c] going back to pioneering works such as those of Lionel March and Philip Steadman

a http://www.spacesyntax.net/

b http://www.bartlett.ucl.ac.uk/casa

c To keep up with updates in this field we have been checking this extensive database: http://cumincad.scix.net/cgi-bin/works/Home and followed the publications of the Pion Publishers and relevant articles of the journal Environment and Planning B[a].

We have also checked relevant topics from the scholarly journals on Geographic Information and Geomatics[b] where on urban networks and accessibility. We have searched for papers also in the areas of design, planning or spatial decision support systems. Literature review has been carried out throughout the research process to keep updated with the state-of-the-art in the aforementioned disciplinary areas. In addition, it is notable that prior to mathematical computational research we skimmed many papers in the interdisciplinary domain area of design methodology and design research represented by key figures such as Herbert A Simon, Donald Shon, Horst Rittel, Nigel Cross, Kees Dorst, Brayan Lawson, Rivka Oxman, and Gabriela Goldschmidt. This was because of the intention of delivering design (research) methodologies that will be practically used, so it was necessary to know more about the design process before thinking of structuring it.

§ 1.10.2 Problem Formulation and Conceptual Design

Formulations of what could be or should be designed as 'configuration analysis and synthesis' methods in architecture and urban design have been revised in iterations after each feedback cycle of test and development. The theoretical underpinnings of this research are rather phenomenological in line with those of Space Syntax; however, we have eventually developed alternative or complementary methods to those of space syntax. We have devised a conceptual framework that consistently deals with architectural and urban spatial networks, which is implemented and tested as prototypical spatial decision support systems. Based on the mathematical framework, we have developed data models, algorithms, and processes (design or analysis workflows) for dealing with architectural and urban configurations.

The most notable mathematical and/or computational constructs developed in this research are as follow:

- the Easiest Path algorithm,
- Fuzzy Spatial-Temporal Accessibility Models (of walking and cycling),
- generalized Network Centrality Models,
- Graph Drawing Methods,
- Generalized Voronoi and Alpha Shape models (of walking/cycling zones),

a http://www.envplan.com/B.html

b Such as, but not limited to the journal of Geographical Information http://onlinelibrary.wiley.com/journal/10.1111/(ISSN)1538-4632

– Interactive Bubble Diagrams,
– A Plan Layout Topology Enumerator,
– Isovist Bubbles, Easiest Path, and
– A family of Random Walk models (Discrete Time Markov Chain [DTMC]).

§ 1.10.3 Implementation and Test Cycles

In order to ensure the methods and models perform consistently in mathematical terms, many verification tests on the implemented results have been performed urging new developments of methods, models, and algorithms. Methods presented in this dissertation have been programmed and put in the form of design toolkits. These toolkits, have been released publicly as freeware applications, and have been tested ever since, by two growing and diverse user groups dedicated to dissemination of the design toolkits and collecting user feedbacks. Here we introduce the toolkits briefly and provide links to their download pages and their user groups:

SYNTACTIC	
Space Syntax for Generative Design (2013 onward); features include real-time space syntax centrality indicators, interactive bubble diagrams using automated graph drawing, revealing all possible planar topologies for plan layout, and Isovist Bubble Agents **Dedicated User Group:** Space Syntax [for Generative Design]	SYNTACTIC
CONFIGURBANIST	
Urban Configuration Analysis for Walking and Cycling (2012 onward): features include computation of easiest paths, fuzzy accessibility measures, walking and cycling zones of preferred access using Generalized Voronoi Diagrams and Alpha Shapes **Dedicated User Group:** Cheetah the CONFIGURBANIST	CONFIGURBANIST
CONFIGRAPHICS	
Configraphics.dll is a library of methods used in both SYNTACTIC & CONFIGURBANIST for analysis and synthesis of spatial configurations[8].	forthcoming

§ 1.10.4 Research Tools and Techniques

Mathematical and Computational Modelling and Simulation have been the main tools of this research. The tool suites CONFIGURBANIST[a] and SYNTACTIC[b] contain [partial] implementation of the methodologies presented in this dissertation. They have been developed for Rhino3D®[c] and Grasshopper3D©[d], released and tested since 2012. Algorithms have been implemented in C# and VB.NET (Object-Oriented Programming languages from Microsoft Dot Net Framework, alias MSDN) in some cases using type definitions and geometric computation methods of Rhinocommon[e], i.e. the library of methods provided by McNeel[f] company, i.e. the vendor of Rhino3D and Grasshopper3D. Urban network models are extracted from OpenStreetMaps[g] or governmental 2D and 3D GIS data.

§ 1.10.5 Assessment and Adaptation

There have been many implementations, which we do not report in this dissertation for they have been unverified or invalidated in cycles of test and development. The last implementations have been sequentially tested in international workshops and then released to computational design community to benefit from crowd sourced test and validation of intended features.

The international workshops conducted by the author (together with international colleagues) are enlisted below in chronological order:

- Measuring Urbanity[h], Lisbon 2012 Technical University of Lisbon, with Dr. Jose Nuno Beirao, Dr. Jorge Gil, Dr. Nuno Montenegro

a https://sites.google.com/site/pirouznourian/configurbanist
b https://sites.google.com/site/pirouznourian/syntactic-design
c https://www.rhino3d.com/
d http://www.grasshopper3d.com/
e http://4.rhino3d.com/5/rhinocommon/
f http://www.en.na.mcneel.com/
g https://www.openstreetmap.org/#map=15/52.0085/4.3693&layers=CD
h http://www.measurb.org/en/home.html

- "Tarlabasi" Datascope[a], Istanbul 2013, Istanbul Technical University, with Dr. Ahu Sokmenoglu and Dr. Jose Nuno Beirao
- Urban Datascope[b], Delft, at 31st eCAADe conference 2013, with Dr. Ahu Sokmenoglu and Dr. Jose Nuno Beirao
- Generative Syntax in Architecture and Urban Design[c], with Ir. Richard Schaffranek at AAG 2014 (Advances in Architectural Geometry, UCL, London)
- Cityscape Configuration[d], with Ir. Philip Belesky at 33rd eCAADe 2015, TU Wien

§ 1.11 Related Tools and Methods

The following software applications are related to the area of this research in that they provide various methods of configurational analysis or synthesis. The actual list could be longer, but these are representative (not exhaustive):

- **DepthMapX** Space Syntax and Visual Graph Analysis developed by Alasdair Turner, maintained by Tasos Varoudis at UCL
- **CASA space syntax[e] software by Michael Batty**
- **Urban Network Analysis (UNA)** Toolbox for ArcGIS 10.1 and Rhinoceros 5 by Andre Svetsuk and the City Form Lab & Urban Network Analysis Toolbox for Rhino3D[f] by Andres Sevtsuk, Michael Mekonnen, Raul Kalvo.
- **Space Syntax for QGIS**[g], by Jorge Gil, UCL Space Syntax Laboratory
- **Place Syntax[h] by Alexander Stahle et al KTH**
- **SpiderWeb** for Grasshopper[i] by Richard Schaffranek a graph theory algorithm library for generative design (presented in (Schaffranek, R. and Vasku, M, 2013))
- **Decoding Spaces** suite for Grasshopper[j] by Martin Bielik, Sven Schneider, Reinhard König, not currently available

a https://tarlabasidatascope.wordpress.com/
b https://urbandatascope.wordpress.com/ & http://ecaade2013.bk.tudelft.nl/
c http://www.gbl.tuwien.ac.at/Archiv/digital.html?name=AAG2014
d http://info.tuwien.ac.at/ecaade2015/workshops
e https://www.bartlett.ucl.ac.uk/casa/latest/software/ajax-software-for-generalised-syntax
f http://cityform.mit.edu/projects/urban-network-analysis.html
g https://github.com/SpaceGroupUCL/qgisSpaceSyntaxToolkit
h https://www.arch.kth.se/en/forskning/urban-design/spatial-analys-and-design-sad-1.298350
i http://www.gbl.tuwien.ac.at/Archiv/digital.html?name=SpiderWeb, http://www.food4rhino.com/project/spiderweb?etx
j http://www.decodingspaces.de/content/decoding-spaces-components-grasshopper-rhino

- **Place Logics[a], an urban centrality analysis methodology (Sergio Porta et al)**

a http://www.placelogic.org.uk/insights/place-logic-brochure-2015.html

TOC

§ 1.12 Scientific and Societal Relevance

The key areas of scientific contribution of this work are advance theories and methods of spatial analysis for studying 'movement and social interaction potentials in buildings and built environments'. Specifically, this project proposes:

- methods for synthesising spatial layout based on configuration accompanied by real-time feedback on spatial performance; and
- methods for analysing 'the effects of configuration on active mobility (i.e. walking and cycling)', while introducing new methods and algorithms such as Topological Layout Enumeration, Isovist Bubbles, the Easiest Path algorithm, etc.

The proposed mathematical-computational frameworks will facilitate further systematic design research in the fields of Computer-Aided Architectural Design (CAAD) and Spatial Decision Support Systems (SDSS).

The societal relevance of the proposed architectural design methodology is to provide a workable way of studying the potential programmatic performance of complex buildings, while offering an intuitive configurational approach to spatial layout. These methods can be eventually used in optimizing the functioning, as well as, safety and security of buildings such as hospitals, airports and alike. The urban configuration analysis methodology developed in this research provides a comprehensive framework for assessing feasibility and suitability of walking and cycling according to the geometry and topology of environment and the location of facilities (land-use attractions). It therefore helps in assessing scenarios in which walking and cycling as sustainable modes of mobility are to be promoted, by providing the essential knowledge base for spatial analysis and prediction of possible outcomes of any action that would change the configurational structure of built environment.

§ 1.13 Technology Readiness Level

The methodologies have been implemented as design technologies, which are already available outside the lab environment. These developed technologies range in TRL (technology readiness level) of 4 to 7 (source: HORIZON 2020 – WORK PROGRAMME 2014-2015 General Annexes). The definitions of the TRLs are given below for reference:

- TRL 1 – basic principles observed
- TRL 2 – technology concept formulated
- TRL 3 – experimental proof of concept
- TRL 4 – technology validated in lab
- TRL 5 – technology validated in relevant environment (industrially relevant environment in the case of key enabling technologies)
- TRL 6 – technology demonstrated in relevant environment (industrially relevant environment in the case of key enabling technologies)
- TRL 7 – system prototype demonstration in operational environment
- TRL 8 – system complete and qualified
- TRL 9 – actual system proven in operational environment (competitive manufacturing in the case of key enabling technologies; or in space)

§ 1.14 Outline of the Dissertation

Chapter 1 gives an overview of the dissertation. Chapter 2 goes in depth in theoretical underpinnings of the research and the research methodology of the thesis. Chapter 3 presents models and methods developed for analysing and synthesising architectural configurations. Chapter 4 presents the implementation results and tests performed on the methods of chapter 4 in SYNTACTIC tool suite. Chapter 5 presents the structured collection of models and methods for analysing urban configurations, computation of easiest paths for walking and cycling, fuzzy accessibility and catchment and construction of a mathematical model of a Markov Chain for probabilistic modelling of expected value of pedestrian or cyclist flows. Chapter 6 presents the implementation of methods of chapter 5 in the tool suite CONFIGURBANIST. Chapter 7 concludes the research by summarizing achievement, contributions, and limitations of the research results. Figure 2 shows an outline of the dissertation.

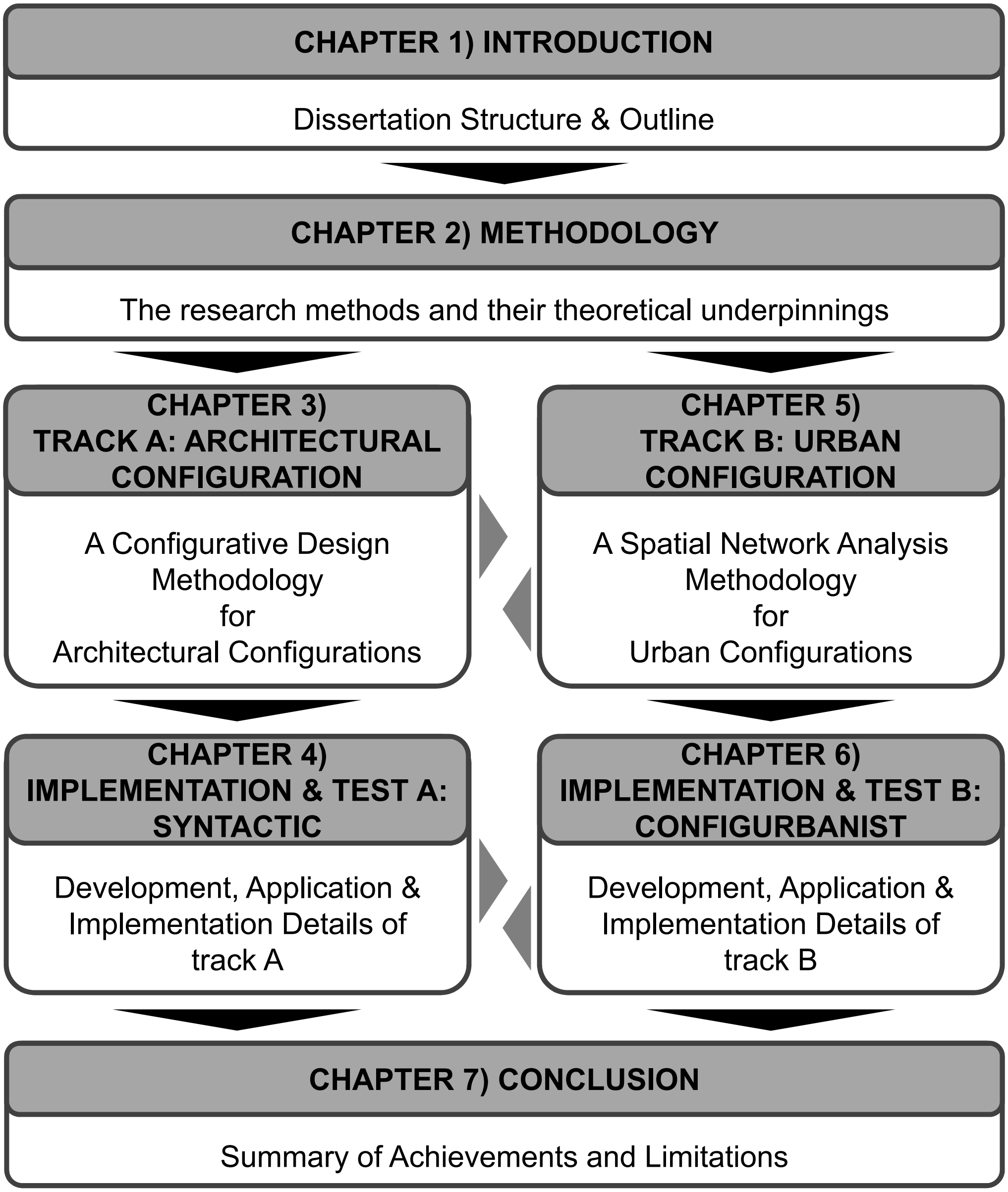


FIGURE 2 dissertation outline

2 Research Methodology

What distinguishes a research methodology from a mere collection of methods is a set of theoretical underpinnings, which can be considered as the theory of the methods (Horvath, 2004). This chapter is to present the theoretical underpinnings of the work and explain its research methodology. Additionally, we clarify our view on the theoretical extent to which mathematical and computational models, analyses and simulations can benefit planners and designer in assessing the consequences of spatial decisions in spatial configurations such as 'large & complex buildings' and 'cities'.

§ 2.1 Introduction

This chapter eventually presents the way the products (models, methods, tools, etc.) of this project have been developed. Prior to that, it elaborates on the theoretical underpinning of this work; especially regarding design methods, necessity of methods, systematization of design processes, and the role of analytic tools in design.

It is obvious that the methods used to develop the products of this research are mainly of mathematical and/or computational nature. Had the definitions existed prior to the commencement of this research, the project would have been a relatively straightforward engineering R&D project in the areas of computational graph theory, computational topology[a], computational geometry, computer aided geometric modelling, mathematical modelling, and scientific computing. However, the very definition of 'what' needed to be developed has been one of the key subject matters of the research; in fact, we have revised such definitions several times. In the following sections, we give an overview of our theoretical reflections on 'what needs to be developed'. Afterwards, we give an articulated description of 'how the models, methods, and constructs have been developed'.

a Computational Topology (Edelsbrunner, H., & Harer, J., 2008) deals with General Topology (Point Set Topology) and particularly with Algebraic Topology which deals with simplicial complexes and Topological Graph Theory (that studies the spatial embeddings of graphs). Hereafter, for brevity and simplicity we refer to these fields as topology with the suffixes .ic or .ical for adjectives refereeing to constructs pertaining to these fields.

Readers who are only interested in 'how the work is carried out', might skip all the following sections and go directly to the section § 2.7. *Methodological Approach*, i.e. where we elaborate our earlier description of the research methodology in the previous chapter. The sections preceding that will explain reasons behind 'why the work is carried out in this way'.

§ 2.2 Background & Definitions

As mentioned above, having had a clear-cut definition of what needed to be developed at the beginning, the research could have been potentially reduced to a relatively straightforward engineering research. However, a great deal of effort has been put exactly on defining 'what' should be developed, regarding its ultimate application in design. The nature of design activities in architecture and urban design is different from most engineering design tasks in that these disciplines require performance criteria pertained to human behaviour that are not easily measureable. The most important functional requirements for designing buildings and cities pertain to human factors and social behaviours, as in large complex buildings and cities. Conversely, in most other engineering fields design requirements can be expressed clearly and undisputedly in terms of physically quantifiable performance criteria. The inherently complicated nature of design problems -in architecture and urban design- brings about a number of challenges in approaching design as a systematic activity. However, the very idea of a design method implies a view of design as a potentially systematic activity. Here we report some key points from our design research studies and reflections on these matters, namely on 'the nature of design activities' and 'systematization of design'.

§ 2.2.1 What is special about design problems?

In an engineering design task, for instance in designing a structure, such things as minimum use of materials or maximum seismic resilience of the structure are key performance aspects, which can be defined, analysed, and evaluated; based on which design options can be objectively compared so as to optimize a design. This is not the case for architecture and urban design in that their essential functioning has to do with the way people would interact with them. People as intentional/anticipatory systems are not predictable as natural causal systems; at least not so certainly as natural systems. This is one of the reasons as to why humanities and social sciences often employ alternative forms of enquiry, which are essentially different from those of natural sciences.

Design and Planning problems have been distinguished from engineering problems as ill-defined/ill-structured problems (Simon, 1999), wicked problems (Rittel, H. & Webber, M., 1973), unique problems (Schon, 1987), or situated problems (Dorst, 2007).

In their seminal paper 'Dilemmas in a General Theory of Planning', Rittel & Webber(ibid) describe the peculiarities of planning problems in terms of the absence of consensus on definitions of quality criteria, the vagueness of problems themselves, the unknown ways to solve them and alike. Most of such difficulties can be traced as to be about the human aspects of design and planning issues, simply because, in design and planning, we are not dealing merely with physical measureable things, but we are also dealing with social debatable aspects of things. We are designing for human societies whose views regarding what is of quality are usually subject to social, cultural, political, or economical debates. The geographical, contextual, and societal uniqueness of design problems is remarkable. Rittel and Webber term design and planning problems as 'wicked', in order to emphasize the fact that they elude easy formulations: both at the level of goals and at the level of their variables. Most important of all, they mention that these are problems for which one could conceive 'good or bad' solutions but not 'true or false' ones. We can realize this fact by looking at all the debates going on about the way we can measure spatial qualities; or more controversially, if we can measure sustainability in its entirety and alike. From a philosophical point of view, this implies that arguing about 'absolute optimality' of a design solution is rather futile or pointless (ibid).

To deal with design problems, designers have to 'frame' a situation and 'formulate' design requirements into what we can formally call 'design problems'. It is deluding to jump to the conclusion that design is about solving such problems; while the more important step in design is the very formulation of a problem (elaborated in the section§ 2.3.1 Design Paradigms); as a problem formulation determines the scope of what 'can be possibly achieved' out of a design process. To illustrate this point, we shall contemplate on a thought experiment on 'problem solving' later in the next section. Design problems are also characterized with vague and (often)-conflicting goals, whose importance are usually hierarchical. It would be simplistic to deem design as seeking for the best option, as there are usually no benchmarks clearly defining what could possibly be the best outcome, let alone the complicated task of creating options to begin with. In many situations, design is practically about explorations and making trade-offs so that a performance aspect becomes 'good enough' whereas another is as good as possible within the (often many) constrains. All such evaluations are also subject to an evolution of the formulation of problem itself; as the definition of what is good also evolves throughout an exploratory process (Designing without Final Goals, (Simon, 1999, pp. 162-165)), and (Dorst & Cross, 2007).

§ 2.2.2 Problem Formulation vs Problem-Solving

There is certainly room and necessity for many optimizations in any design process, but the whole process cannot be reduced to an engineering problem solving (optimization) process for two main reasons: firstly, because such a view takes the existence of problem formulations for granted while disregarding the complicated reality of designing. Suppose a hypothetical situation in which a designer is to design a polygonal house plan to be built with a certain limited amount of bricks in order to maximize the house area, choosing a rectangle as a basic form, we can immediately jump to optimization and formulate the problem as follows:

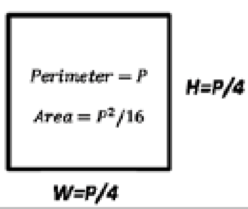
Problem Setting/Formulation	Suppose the design is formulated as a rectangle with the width W and height H, which its area is desired to be maximized (Given the perimeter as a constant P). In other words, the problem is to find the maximum rectangular area that one can circumscribe with a rope of the length P. We have: 
Constraint	$P = 2(W + H) = Constant$
Design Variable	Either W or H can be considered as a variable parameter: $Either \quad H = \frac{(P-2W)}{2} \quad or \quad W = \frac{(P-2H)}{2}$
Objective (Fitness) Function	We can write the Area as a function of the single variable W as below: $Area(W) = W.H = W.\frac{P-2W}{2} = PW/2 - W^2$
Problem-Solving	$Area'(W) = P/2 - 2W$ $Let\ Area'(W) = \frac{P}{2} - 2W = 0 \xrightarrow{yields} W = P/4\ \&\ H = P/4$ $Area_{max} = W.H = P^2/16$
Solution	

FIGURE 3 an exemplary design optimizatoion probelm

The maximum area achieved here is equal to $\mathrm{W.H} = P^2/16$, whereas if the designer in question had chosen a circle, they would have achieved the following surface area:

$$A = \pi r^2, P = 2\pi r = const. \xrightarrow{yields} A = \pi(\frac{P}{2\pi})^2 = \frac{P^2}{4\pi} > \frac{P^2}{16}$$

This implies that formulating the plan, as a circle would have resulted in a better performance, given this view of utility (maximum possible area). This is to show the importance of the very first formulation. In this case, it is rather easy to see that there is a shape with an absolute maximum surface area, given a constant perimeter (i.e. a circle). However, even this formulation might result in a design that is of little overall utility, because of such things as the difficulty of planning spaces in a circular plan (e.g. because of rectangular furniture, etc.). In other words, it will not be feasible, wise, or practical to formulate all design goals mathematically, and even if possible, then the initial formulation would determine the extent to which a certain design can achieve a certain level of performance. This is to say problem formulation is much more important than problem solving. Note that problem-solving methods are usually routine and standard whereas there are hardly any simple set of explicit rules for formulating design problems. Some might think that Multi-Objective Optimization is a final solution to all design problems. We argue that these are search methods for finding the best option in 'search space' (design space) that the designer has defined. If the designer has not given rise to a set of good options, they will never be found even with the best search algorithms!

§ 2.2.3 On Automated Design

Although not a very popular idea anymore in architecture and urban design, there are still some scholars pursuing the goal of making design machines that could automate design. The matter of automating design is perhaps interesting from the perspective of artificial intelligence research. However, automating design decisions in most cases is not practical. Observe the fact that designers and planners do not use such systems. This is not simply because they fear to lose their jobs as computers might be more intelligent than they are, but mainly because such systems can hardly be used in real design situations. The whole idea of expecting design machines to generate design automagically is so simplistic that is out of question for an experienced designer.

Examples of such systems include algorithms designed to give the best plan layout for an apartment (see for instance this survey (Lobos, D., Donath, D., 2010)). Many such systems define design as minimization of circulation routes and packing; an assumption that is too simplistic for many real-world situations.

§ 2.2.4 Logical Leap in Design

From a philosophical point of view, i.e. Philosophy of Science and Technology, there appears to be a 'logical leap' in design processes (Kroes, Peter and Meijers, Anthonie, 2006). This is because designers are supposed to produce a concrete form in order to fulfil some abstract functions (ibid). Practically, a designer is usually briefed with a verbal vague description of what is required or desired and then they are to produce a spatial structure, which supposedly fulfils those needs. There is no one to one relation between **forms** and **functions** and so it appears that there is a lot of guesswork, reasoning leaps, and speculation involved. In other words, the process does not seem to be particularly logical.

Nigel Cross (Cross, 1982) provides a plausible description of this process, i.e. the transition from abstract to concrete, in terms of how designers "use 'codes' that translate abstract requirements into concrete objects". This is central to our approach to design as we propose systematic ways of bridging this gap between abstract functional descriptions and concrete geo-spatial structures by medium of 'configuration' and 'configurational analysis'.

§ 2.2.5 Design Methodology & Design Research

Design Research is a disciplined [interdisciplinary] conversation (Cross, 1999). The discipline was initiated around central questions on the ideas regarding systematisation of design (by means of design methods), which began with the advent of computers around 50 years ago and the need to rationally and quickly reconstruct the world after the devastation of the WWII.

There were initially positivistic ideas on systematization and even automation of design; such ideas were later mostly rejected (even by those who found such ideas in the first place such as Christopher Alexander). Later there were trends on studying the nature of design activities as practitioners experience them (Cross, 1982).

Design research can be described as the field of study dealing with (Cross, 1999):

- **Design Epistemology:** that is about studying design knowledge[a] and 'how designers think' (Lawson, 1980) and the nature of their so-called tacit knowledge, knowhow, and experience
- **Design Praxeology:** that is about how designers work in practice, how the design process proceeds and what is/could be the role of design, representations, tools, and media in designing
- **Design Phenomenology:** that is about studying the nature of what is produced in designing, mainly about form and configuration studies (conventionally known as architectural or urban morphology)

All topics are related to the subject matter of this research, because if we are to improve a certain practice, firstly we need to know how it is in reality. We need to know: *what form of knowledge can be useful* (concerning design epistemology); *when [procedurally] it can be useful* (concerning design praxeology); and *in what form should we present information to be best integrated with a design workflow* (concerning design phenomenology)? This is to say mostly we need to have a knowledge on the nature and dynamics of the 'design process' in question. In the following section, we specify the theoretical underpinnings of our work, in terms of our epistemology, praxeology, and phenomenology.

[a] This can be a debateable title as epistemology refers to studying the origin, nature, validity, and limitation of scientific knowledge but 'what designers know' and 'how designers think' are not necessarily the subject of epistemology because design knowhow involves unscientific knowledge as well as scientific knowledge. For an alternative account of this topic, refer to (Horvath, 2004). Our view is that design knowledge should be structured and that we should seek to develop a scientific body of knowledge that could guide design actions.

§ 2.3 Theoretical Underpinnings

We advocate a direction in design that seeks scientific knowledge of social and environmental consequences of spatial decision in order to base design decisions on such knowledge models and evidences from spatial analyses.

We do not pursue such goals as predicting the exact behavioural patterns of people in buildings or cities; nor do we advocate replacing observation and reflection by simulation. Simulation models[9] are based on reduction and abstraction and so they cannot replicate the entirety of reality, so they are all wrong[a] in the absolute sense of the word; however, we can discuss the extent to which our models can be useful.

In this section, we specify our view of the nature of design knowledge, design processes, and design representations and theoretical limits on how these can be possibly structured, strengthened, or systematized by means of mathematical and computational models and methods.

§ 2.3.1 Design Epistemology: Design Paradigms

The subject of epistemology is the nature of knowledge and its basis, validity, and limitations. The topic of 'nature of design knowledge and design activities' has been scholastically debated for over forty years so far (Cross, 2007); and there is a paradigmatic discourse named Design Research around it (Cross, 1999)[10]. There have been many efforts on rationalizing design activities; and as one can imagine, many reactions against such efforts. Design Methodology, as a genuine branch of design research is mainly about design processes and the ways they can be organized and improved.

There are two opposing 'design paradigms', which seek to describe the intellectual nature of design activity, while offering propositions for improving it: One emphasizes the importance of creativity and experience (Schon, 1987), while the other is more concerned with the soundness of a design process in terms of scientific research methodology and rationality (Simon, 1999). We can identify a third stance in between, considering both stances as contextually relevant, by deeming design activity as something dependent on the kind of design problem at hand and the level of expertise of the practitioner.

a "Essentially, all models are wrong, but some are useful." George E. P. Box (Box, G. E. P., and Draper, N. R., (1987), Empirical Model Building and Response Surfaces, John Wiley & Sons, New York, NY.)

In Table 1 we present an outline of design paradigms, in relation with their epistemological stances after (Dorst, 1997), (Okasha, 2002), and (Dorst, 2007).

SUBJECTIVITY	INTER-SUBJECTIVITY	OBJECTIVITY
Context of Discovery	Context of Interpretation	Context of Justification
Arguments based on experience and/or observation	Inference to the Best Explanation	Arguments based on rules, laws and widely accepted principles
Design as a Reflective Practice The Reflective Practitioner (Schon, 1987) Unique Problems Brief: Designer is free to 'see' the problem at hand and 'frame' it in a particular manner; based on this frame s/he makes a move and then again 'see's or tests the outcomes in a rather cyclic manner.	Design as Co-evolution of Problems & Solutions (Maher, 1996)& (Dorst & Cross, 2007) Dealing with Wicked Problems (Rittel, H. & Webber, M., 1973) [through] 'Situated Interpretations' (Dorst, 2007)	Design as Rational Problem Solving The Sciences of the Artificial (Simon, 1999) Ill-Structured Problems Brief: Designer can 'reduce' and 'formulate' a design problem using mathematical formalisms and then seek for a scientific 'problem solving method' in a rational manner.

TABLE 1 An outline of design paradigms, in relation with their epistemological stances after (Dorst, 1997), (Okasha, 2002), (Fallman, 2003), and (Dorst, 2007)

The paradigm "Design as a Reflective Practice", maintains that a designer 'sees' the problem; 'frames' it according to their own constructs; makes 'moves' (offers tentative solutions for the at hand problem); and then 'tests' the alternatives according to some criteria and this rather cyclic activity continues till the designer becomes satisfied with a solution. The opposing paradigm, "Design as Rational Problem-Solving", holds that the design problem should be formulated in mathematical formalisms through scientific reduction, and based on such formulations relevant problem-solving methods should be found and applied.

Whether we call it 'formulation' or 'framing', there is a phase of '**interpretation**' at the beginning of a design process. Kees Dorst in (Dorst, 2007) describes the epistemological basis of design as being hermeneutical or interpretational: as the specific kind of reasoning applied to design appears to be so. He maintains that hermeneutics of Hans Georg Gadamer, can bridge the epistemological gap between the opposing poles (of the paradigmatic spectrum) as it identifies "situated interpretations".

That is to say, interpretations in different design situations depend on the nature of the design problems involved in the design assignment. In other words, our interpretation of a design situation, depending on how "determined (i.e. structured and clear)" it is, can be objective, subjective, or inter-subjective. If the nature of design assignment is "determined", meaning that we deal with a *reasonably known generic problem*, it is worthwhile to seek for a generic formulation of it (such as the problems we formulate in this dissertation). Whereas, if the situation is so unique that its peculiarities cannot be reduced into any sort of known formalized problem, or they simply are not worth modelling, then the designer is free to choose a creative frame to interpret the problem in their own way. The importance of designer's interpretation of a situation is clear in both of the opposing paradigms. The subjective paradigm (Schon, 1987) holds 'framing' as the important interpretational activity, i.e. the way a designer 'sees' the problem. With a different terminology, the objective paradigm (Simon, 1999) also considers problem setting as the interpretational activity central to design thinking.

An interpretational notion of design process is constructive because it allows for explaining the kind of actions designers and planners actually take in dealing with various built environment design or spatial planning problems.

§ 2.3.2 Design Praxeology: Design Process & Design Methods

This section pertains to design praxeology as explained earlier. Engineering design methods are used extensively in many fields of engineering from mechanical engineering to software engineering, without much resistance from practitioners. However, proposing design methods for spatial design and planning is a very delicate matter as to the peculiar nature of the 'problems' dealt with in such fields.

Proposing a model (conceptual framework) of design process has been central to design methodology ever since the discipline has come into existence. The two design paradigms define design process in a relatively ideal form; but from our pragmatic point of view, the description of Lawson (Lawson, 1980) turns out to be more useful and accurate, at least about the type of design that we are advocating (i.e. systematic and evidence-based). He describes design as being about Analysis, Synthesis (generation of design alternatives)[11], and Evaluation actions in cycles, going back and forth in between alternating problem formulations and solution propositions, pointing to the fact that design process is not a linear process; but rather re-iterative (in a manner of speaking).

This view is also in line with viewing design as a course of '*co-evolution of design problems and design solutions*' proposed by Maher & Poon (Maher, 1996) , which was later substantiated by protocol studies by Dorst and Cross (Dorst & Cross, 2007). The concept is that design is a course of actions that produces tentative problem formulations and corresponding tentative solutions; the outcomes are the fittest, which have survived tests and evaluations. As the term evolution suggests, there is something about checking 'fitness' of a design in this process. For the reasons mentioned before, mainly that spatial design and planning deal with social, cultural, political and economic systems, it might be quite problematic to talk about 'fitness' in an absolute engineering sense. For this reason, it can be seen that evaluation, i.e. reaching to conclusions on goodness or fitness of a design or plan could be (and perhaps should be) a debateable action. Think of defining such concepts as sustainability (in its entirety) and formulating their meaning in quantitative terms for instance to realize the importance of intellectual debates in evaluation. Synthesis of form and configurations is of course obviously a design specialism. In this dissertation, we shall introduce methods for automated analysis (of spatial configurations), interactive synthesis (for architectural plan layout), and an exemplary way of evaluating walking and cycling accessibility of neighbourhoods.

In line with the abovementioned notions of design process, Nigel Cross has characterized the "designerly ways of knowing" in (Cross, 1982). He maintains that designers have a particular way of thinking, different from those of scientists or artists:

- Designers tackle 'ill-defined' problems.
- Their mode of problem-solving is 'solution-focused'[12].
- Their mode of thinking is 'constructive'.
- They use 'codes' that translate abstract requirements into concrete objects.
- They use these codes to both 'read' and 'write' in 'object languages'.

In a sense, our work can be seen to be in the direction of strengthening this language of 'codes' by benefiting from representations of graph theory.

§ 2.3.3 Design Phenomenology: Design as Spatial Configuration

This section clarifies our phenomenological stance on design. To this end, we focus on the nature of design representations. We can observe a design process by tracing its evolving representations in a spectrum of abstract to concrete: from abstract verbal descriptions of programmatic or functional requirements to concrete physical plans. Spatial Configurations lie somewhere in the middle of this spectrum, in that they can help in defining relations among functional entities as abstract graphs and when embedded in 3D space as topologies they come closer to the concrete geometric world. It is because of such properties that configurations seem ideal for bridging 'the logical leap' in spatial design.

Inspired by the theoretical framework of Space Syntax (Hillier, B., Hanson, J., 1984) theories, we define configurations as a labelled [possibly directed] simple graphs (as in graph theory) composed of nodes and links representing spatial connections between the nodes, be it room-like spaces in buildings or streets in cities. In producing such graphs, certain geometrical or geographical characteristics of spaces can be attributed to either nodes or links of configuration graphs. Units of space in a configuration graph can be rooms (as convex or star-convex spatial units) in buildings or streets in cities. As apparent in this definition, we extend the definition of 'space' to 'geographical space', which can be dubbed 'place' in urban studies. This is where our approach differs from that of Space Syntax and comes closer to that of Place Syntax (Ståhle A., Marcus, L. and Karlström, A., 2008).

Configurations can be analysed in terms of their likely effects on such things as probability of social encounters and interactions, wayfinding, mobility, and accessibility. A configurative approach to design can help in bringing analysis of functional/programmatic performance to design. Such an approach would be intuitive for human designers and at the same time clearly structured and comprehensible for computer programs.

§ 2.3.4 Causality, Limitations of Models and Decision Support

Design is essentially concerned with 'how things ought to be' and characterized by synthesis (i.e. making and composing), whereas science is more concerned with 'how things are' and it is characterized by analysis (see (Simon, 1999, pp. 4-5) (Cross, 1982)). Nevertheless, immediately after something is realized as a physical structure it can be analysed in terms of its 'state', 'behaviour', or 'performance'. However, it is important to have a body of scientific knowledge that could 'explain' how a spatial system would 'probably' work in reality; e.g. in terms of facilitating, encouraging, or hindering certain spatial interactions. Such a body of knowledge can form the basis for a planning support system (PSS) or a spatial decision support system (SDSS)[13].

There are theoretical limits on what we can possibly model, explain, and predict using mathematical and computational models. These limitations mainly arise from a difference between natural phenomena and artificial phenomena.

We need to highlight a difference between the natural environment (mere physical) and the artificial environment (physical environments as spatial systems together with their human inhabitants and designers); which is the fact that a natural system is 'causal[14]' and thus inherently predictable, meaning that their functioning can be modelled in terms of causes and effects. However, artificial systems (such as buildings and cities) have a dual nature in that they obey the laws of physics and at the same time reflect intentions of humans that are not necessarily predictable as 'states' of natural systems, the individual humans, societies and political powers in fact have certain ideas about what/how they want to be in future. This anticipatory nature distinguishes artificial systems from causal systems. This view is adopted from (Kroes, Peter and Meijers, Anthonie, 2006) and (Portugali, 1999).

TOC

For further in depth discussion on the principal limits on modelling and simulation of cities we refer the reader to (Batty, M.& Torrens, P, 2001). The dynamics of a city cannot be simply predicted as the dynamics of a physical system because there can be unforeseen interventions made by human decision-makers, those who might not particularly act according to the self-regulatory processes of evolution and growth[15]. This, however, does not mean that there are no rules governing the form and functioning of cities and buildings. There is in fact evidence that in aggregate scales, some cities have recognizable patterns in the scalar relations between some of their features[a].

§ 2.3.5 Analysis vs Evaluation

Measuring, analysing or estimating the performance of a design is one thing and evaluating its performance is quite another. Evaluation requires a synthesis of many analyses and a framework as to which performance measurements could be judged regarding their relative goodness. This might be done according to performance benchmarks, standards and alike.

We have decided to leave design evaluation methods out of the scope of our proposed methodologies[b] for the following reasons. First, judging the configurational quality of a design is a broad subject, which inherently tends to be context-based. Secondly, we do not intend to replace human interpretation and decision-making by automated procedures. In other words:

- Comprehensive design evaluation has to address social aspects of artificial environments, which are subject to debates, thus necessitating a collaborative or participatory approach for reaching a consensus on an interpretation of qualities.
- Qualitative design evaluation requires contextual information that is not necessarily encoded in the design brief or not represented on maps. This is not exactly because of negligence of designers; there are qualities that are not easily representable or measureable. There will always be need for expert view and involvement of real stakeholders and thus the matter of evaluation is handled through discussions.

a This subject falls outside of the scope of this research but readers interested in the matter are referred to the papers that are relevant to the core of this research; but as an example, interested readers can find interesting scaling studies in (Jiang, 2007).

b With the exception of the relatively straightforward evaluation reports representing the aggregate walking/cycling accessibility in neighbourhoods towards several POI

Instead of automated design evaluation, we thus focus on facilitating design evaluation by proposing methods and workflows for measurements for performance analysis.

§ 2.4 On Computational Design

We exploit computation in support of three types of design activities, namely spatial analysis, spatial synthesis, and spatial evaluation; i.e. following a approach called Performance-Based Computational Design (Sariyildiz, 2012), in which computation is primarily used to systematically address performance in design process. However, because of the multitude of reasons mentioned before, we do not advocate creating automated closed loops of 'design generation and performance evaluation'. Instead, we focus on the use of computational approaches in structuring spatial analysis and layout. The following sections reflect on some practical aspects of computational design tools that can be exploited to this end.

§ 2.4.1 Computer Aided Design (CAD) and Parametric CAD

Old-style Computer Aided Design (CAD[16]) environments provide their functions by means of a command-line and a set of iconic buttons, but do not have the appropriate means for practicing computational design[17]. Computational design can effectively be practiced by using the so-called Parametric CAD environments[18] that provide the means for Flow-Based Programming[19]. We see the emergence of these environments an excellent opportunity, which provides for rationalization of design processes by bringing a number of revolutionary advantages, namely:

- real-time analysis can be performed;
- bespoke quantitative assessment methods can be made
- design process will be explicit and thus more prepared to be rationalized;
- design alternatives will be flexible and rather easily adjustable;
- collaborative design is facilitated greatly;
- design process can be designed and systematized;
- design tools can be extended or improved by scripting;
- designers can customize their design tools easily; and
- by keeping the rationales the same, a variety of options can be generated, i.e. a 'design space' can be explored.

In addition to all these benefits, certain procedures, which were bounded to GIS applications, such as connection to geo-spatial Database Management Systems (DBMS) and processing of geo-data, are also possible in parametric CAD environments and web-based programming platforms. This has an important implication for the future of design and planning practices: it is a recognized issue, that producing spatial designs and plans is mostly done within CAD environments, whereas spatial analysis is traditionally done within GIS environments or standalone applications such as Depthmap (for Space Syntax Analysis). Therefore, in order to bring spatial analysis to 'design process' we naturally chose Parametric CAD environments to put forward our design methodologies.

§ 2.4.2 Design Space Exploration

Computational synthesis allows for exploring multiple design options with a single set of design rationales. The set of all possible design options encoded in a computational model is technically referred to as a 'design space'. We shall see an example of such process in the next chapter in finding all possible plan layout topologies, which correspond to a single bubble diagram (as in an adjacency graph). This way, the spatial layout process enumerates alternative designs. If all options are enumerated, then the designer can choose the ones with a better performance by means of ranking (in terms of estimated performance). If multiple objectives are to be taken into account, then a ranking based on Pareto Optimality can be done.

§ 2.4.3 Real-Time Analysis and Geo-Design

The most important advantage of computational design methods is the possibility of integrating real-time analysis engines with the design process. This process in the context of urban design and planning is referred to as Geo-Design and in the context of architectural design as Performance-Driven design. The essence of the idea is that instead of producing a design/scenario in a long design or planning process and then analysing them afterwards in terms of their [likely] performance, analysis can be brought to the very heart of decision making process or integrated with it. Having the possibility of real-time analysis will allow for making better-informed decisions and prevent unnecessary iterations.

§ 2.4.4 Feedback vs Feedforward

A literal adoption of the view of design as "loops of analysis, synthesis, and evaluation" might mislead us in thinking that the whole loop can be closed and automated. This is often a temptation for those occupied with computational design. The first question is, how exactly evaluation can be integrated with a design or planning workflow. The answer to this question regards two distinct control strategies known as **feedforward** and **feedback**. While the latter term is largely known and often misused by non-professionals and even in journalistic context, the former is not so well known.

In view of hierarchical nature of most design problems, it is reasonable to have certain aspects of a design adjusted before going to problems of a lower level of importance. In doing so, there might be room for some partially automated evaluation processes on inherently measureable physical performance indicators of a design. This diverts us to the question on evaluation and its role in a design process. It can be done after generating plenty of options as a means to choose optimal ones (i.e. in a feedback loop) or it can be done at the very beginning of the process to lead the process immediately in the right direction (i.e. nearly a feedforward strategy)[20].

Let us suppose a planning process that is to intensify a neighbourhood by adding new residences while ensuring most new residents have good walking or cycling access to the facilities in the neighbourhood so that they do not impose a heavy car traffic burden on the infrastructure. In this case, we can first generate a number of scenarios and then test them with an accessibility analysis tool or, alternatively, first provide a map of accessibility and then design the scenarios accordingly. The former would be a feedback strategy and the latter a feedforward one. That is to say, the ultimate design can be based on a synthesis of the analyses performed.

§ 2.5 On Spatial Analysis

Spatial Analysis is the broad field of methods for analysing phenomena that have spatial distributions[21]. Spatial Analytic methods and techniques include spatial data queries regarding geometric or topological relations[22], Spatial Statistics, and Spatial Network Analysis methods. We introduce spatial network analysis methods that can be used to support decision-making in plan layout, urban network development, or land-use planning. The analysis of spatial networks mostly falls into the area of graph theory and is influenced historically by methods of social network analysis. However, prior to analysis, there are methodological challenges in modelling spatial networks. This

TOC

subject, i.e. modelling a real world spatial network as a graph, technically falls into the areas of topology and geometry; and at the same time concerns our phenomenological approach to how space and proximities matter in determination of human behavioural patterns. For these reasons, there are different trends and approaches in modelling spatial networks, as discussed in depth in chapter 5 Besides, in architectural layout, conceptions of spatial networks evolve and change gradually. At the beginning of a design process, the idea of a spatial network can be reduced to a set of nodes and links (graph theoretical); however, later we will have a spatial layout (topological embedding); and at the end of the process the design is so concrete (geometrical) that it can be analysed as to visibility of spaces.

§ 2.5.1 What is special about spatial analysis?

Buildings and Built Environments have clear spatial manifestation and thus it seems obvious that we need to study many aspects of their functioning in space. The essential question is how should we represent space? Is it sufficient to add geographical coordinates to whatever we are measuring? Does that make our analysis spatial? We can represent geographical space in multiple ways. In 3D modelling in CAD environments, we literally model objects in Euclidean space with Cartesian coordinates (RxRxR), i.e. also known as R^3. If we use only geographic coordinates (latitude, longitude, and probably altitude) then we are indirectly modelling space as a 'flat' Euclidean space in reference to a topologically quadrangular surface around the globe. This is what we all know as a world atlas. Nevertheless, it is interesting to look at it again in Figure 4. This is to remind us of the fact that the representation of space itself is an abstraction.

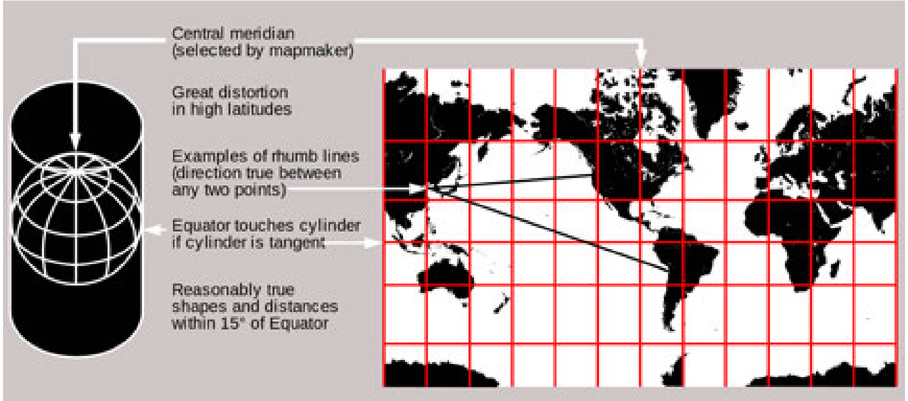


FIGURE 4 Miller Cylindrical Map Projection. A world atlas is a 2D representation of the world corresponding to a rectangular parametrization in terms of latitude and longitude. If we add altitude to the coordinates, as in what we can obtain from GPS devices, then we can enhance this map into a 2.5D representation, in which to every X,Y coordinate pairs we can attribute only one Z coordinate. Image Credit: Wikipedia Commons.

We can also represent the space in terms of quantized units such as pixels or voxels and represent spatial distributions as 'raster' objects. This representation is mathematically equivalent to representing space as the Cartesian product of integer sets such as $\mathbb{Z}\times\mathbb{Z}\times\mathbb{Z}$ that is conventionally shown as $\mathbb{Z}^3$. The geographic space can also be represented as network space or topological space, by virtue of the concept of neighbourhoods in topology. This means that space will be represented as a discrete set of locations where something can be located, e.g. streets or rooms of a building. Such a representation brings a number of immediate advantages, namely, the relative ease of computing geodesics or optimal paths between locations on a network. Such a representation is particularly interesting for spatial analysis because it comes close to our perception of space as in 'being somewhere'. We do not understand our location in space in terms of some numeric coordinates; instead we associate our location with such things as being in the Room A, in Street B, next to Building C and alike. In other words, a topological model of space also provides for establishing 'semantic relations'. This means that using spatial network models we can more easily make [semantic] sense of (potentially big) data attributed to built environments. We go one step further and suggest that disregarding topological relations in spatial analysis diminishes the value of analysis to an extent where the spatial relations can be neglected or miscalculated. We provide the following example for this argument. Suppose in an analytic process -such as clustering rooms based on their accessibility- we considered two rooms as close because their centroids are close to one another in Euclidean space; then chances are, that we are totally mistaken; for the actual distance between two rooms is much longer; in that it is experienced by passing through a possibly much longer way through corridors and other rooms. The same is true and perhaps more obvious for urban studies, e.g. two buildings on the opposite sides of a river or a highway would be close in Euclidean space but probably very far away in terms of network distance.

Supposing that we want to represent space as a network space, then there are other problems and challenges to deal with, most important of which regards the 'units of space'. There are multiple ways of representing spatial units such as convex spaces and axial lines in buildings and axial lines, road centrelines, named streets, etc. in urban studies. We shall discuss these representations compared to our spatial network representations in the chapters 3 and 5. In short, answering 'where' questions or speaking of spatial distributions of phenomena in spatial analysis necessitates a profound approach to both mathematical and semantic implications of different representations.

§ 2.5.2 What-If Scenarios

One of principal uses of spatial decision support systems in planning and design practice is to predict the likely effect of spatial decisions[23], as in changes in the form or functional attributes of environments. Conventionally the notion of what-if scenarios refers to the study of land-use planning options in spatial planning with the aid of Cellular Automata or Agent-Based Models[24] (Portugali, 2006), (Batty, 2007). Nonetheless, if we adopt a broader definition, such systems can be used to assist in making spatial decisions and analysing their effect on the future functioning of a built environment. As mentioned before, we cannot predict - in the rigorous sense of the word- the future states of non-causal systems such as artificial environments, especially because of the role of inherently unpredictable human decision makers (be it inhabitants, planners, or politicians) on the form and functioning of the built environments. Nevertheless, certain aspects of performance of built environments can be modelled to provide insight into what would happen in the absence of radical unpredicted changes imposed to the systems. Besides, taking approaches such as stochastic modelling or modelling uncertainties by means of fuzzy logics or probabilistic approaches we can provide insight into the likely statistical state and performance of such systems. In doing so, providing outmost physical clarity would be very important to avoid the feeling of presenting a 'crystal ball' to designers. That is to say, we need to be clear on what we can model and simulate in terms of physical or measureable quantities that correspond to something concrete. In our approach, we only provide measures that have a tangible physical interpretation and clarify that our models mostly replicate what is possible, but not necessarily, what is probable. This approach is apparent in our modelling of walking and cycling accessibility in chapter 5.

§ 2.6 Highlights

Design and planning professions are about making the artificial environment, that is essentially different from natural environment in that we can associate with artificial objects and environments the functional 'purpose' behind them; and that their state is not essentially predictable as that of natural environments. Design and planning deal with problems that are essentially different from engineering problems in that their problems are inherently ill defined and ill structured, any formulation would be debateable, there is usually a contextual influence, and that the solutions to these problems cannot be evaluated indisputably. Design is about analysis, synthesis (providing alternative plans or designs) and evaluation. Analysis can be automated but evaluation cannot be always automated because of contextual parameters that cannot be necessarily modelled or accounted for. Design and planning processes should thus allow open discussion to facilitate reaching a 'consensus'. A computational design approach can bring real-time analysis to design process to support making better-informed spatial decisions. Although we cannot expect analytic models to predict the exact future state of an artificial environment, in light of the complications inherent to human decision-making, spatial analysis is needed to gain insight on the 'potential performance' of built environments. Without any analytic knowledge, there will be a logical leap in design reasoning, meaning that one could only propose some interventions or constructions only 'hoping' things to function in a certain way.

Specifically, in analysing human activity patterns and mobility in buildings and built environments we need to employ methods and approaches from social sciences such as environmental psychology and social network analysis for modelling and simulation of built environments. We propose that by focusing on synthesis and analysis of configurations, the aforementioned logical gap can be more easily bridged. Our definition of configuration graphs encompasses the previous definitions such as those of Space Syntax theories, while extending the meaning of configuration graphs the inter-relations of what is called geographic space or 'place'.

§ 2.7 Methodological Approach

Here we present an articulated account of the methods used in this research. We initially presented a brief schema of our methodology in Chapter 1/Research Methodology. Figure 5 expands and elaborates that schema (Figure 1). We explained how we have approached our literature studies in the same section in chapter 1. Here we elaborate on some of the actions that require more in depth introduction.

Therefore, we begin by giving an overview of how we went about designing the two main products of this research.

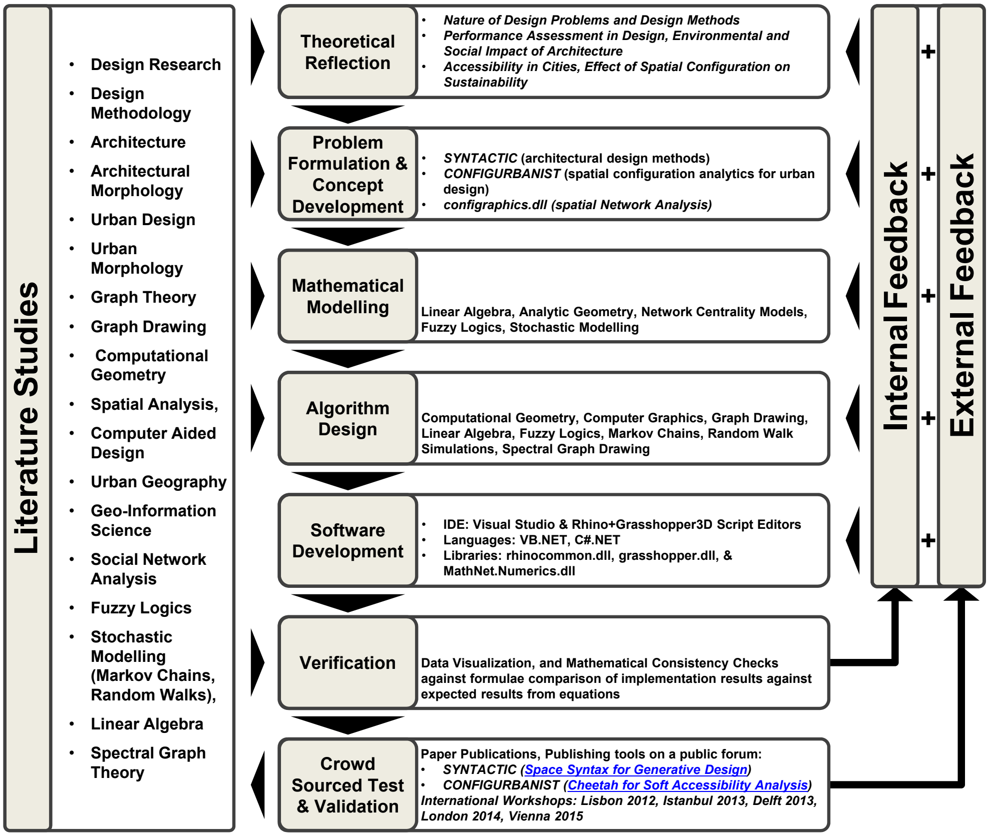


FIGURE 5 An articulated schema of the research, development and innovation methodology of this project

§ 2.7.1 Theoretical Reflection

The technical aspects of this project are easily identifiable; however there has been also many reflections on 'what to do' prior and posterior to 'how to do' those actions. The theoretical reflections have been focused on the efficacy (usefulness) of the mathematical constructs and computational models. It is necessary to refocus continuously on the initial goals of the project along with technical developments. In light of the project goals, theoretical reflections were to ensure the developed constructs would be relevant to actual design practices. Such reflections have resulted in a number of adaptations of the methodical/technical constructs. For instance, we diverted from the technical goal of producing rectangular plan layouts (in the work presented in chapter 3) because we realized in spite of its interesting mathematical concept, the result would have been of little practical interest in real-world architectural design processes. Therefore, we sought a generic alternative, devised new constructs (Isovist Bubbles in 2D and 3D), which could be used in free-form designs. In chapter 5, we have rejected several previous versions of our indicators and sought new ways of formulating accessibility indicators in accordance with intuitive notions of proximity; hence, we decided to use Fuzzy logics to model accessibility as it is perceived by humans; for which we used time as a common denominator in measuring accessibility.

§ 2.7.2 Problem Formulation & Concept Development

The concepts behind the two methodologies implemented in the toolkits SYNTACTIC and CONFIGURBANIST have been formed gradually, while reflecting on the computational design possibilities and the necessities for analytic tools in design processes. Each concept formulation then initiated a number of problems of mostly technical nature (mathematical or computational). The concepts of both methodologies have been revised and redeveloped multiple times ever since based on the internal and external feedback loops illustrated above. It is not always easy to foresee what can be possibly achieved with a concept, unless it is developed as a concrete product.

§ 2.7.3 Mathematical Modelling

Graph Theory was chosen at the very beginning of the project as the main mathematical field of study. Linear Algebra and Analytic Geometry are obviously needed in dealing with Computational Geometry and Computer Graphics. Fuzzy Logics (Zadeh, 1965) was a clear choice for modelling the perception of people of such concepts as distance as it is meant to help in mathematical modelling of language variables and verbal concepts. After theoretical reflections on the nature of human behaviour in space, we decided that stochastic models would be useful in partially explaining patterns of mass movement in urban spaces, inspired by the approach of (Blanchard, Philippe, and

Dimitri Volchenkov, 2008). Moving in that direction demanded Spectral Graph Theory as a basis for constructing our Markov Chain (Random Walk) models.

§ 2.7.4 Algorithm Design

Several models and methods presented in this work have mathematical formulations but their solutions can be found only computationally, i.e. they do not have analytic solutions. In other words, they only have algorithmic solutions. Key examples are:

- Topological Modelling (i.e. constructing mathematical graph representations out of collections of points, lines, polygons, and polyhedrons, based on Poincare Duality Theorem, e.g. Graph Models in Chapter 3 and Chapter 5)
- Graph Traversal (e.g. Breadth-First-Search & Depth-First-Search),
- Path Finding algorithms (e.g. Dijkstra Algorithm & Floyd-Warshall Algorithm),
- Graph Drawing Algorithms (e.g. Tutte Convex Drawing and Force-Directed Graph Drawing in Chapter 3),
- Enumeration of Triangulation Patterns of a Polygon (in Chapter 3, enumeration of all possible plan layout topologies, based on a formulation given by Leonhard Euler),
- Computational Geometry, e.g. 2D&3D Isovist Bubbles (Smart Star-Polygons, which maintain their visibility and area in presence of obstacles)
- Fuzzification Algorithms (as in fuzzification of closeness measures in Chapter 5)
- Fuzzy Aggregation Models (i.e. Zadeh, Yager ,and Paraboloid Fuzzy Logical models for AND, OR aggregations of closeness values)
- Computational Topology (e.g. in Weighted Voronoi Diagrams in Chapter 3 and Generalized Alpha Shapes in Chapter 5)
- Algorithmic Linear Algebra (e.g. in our Generalized Power Iteration Method for finding eigenvectors)
- Algorithmic Linear Algebra in finding Stationary Distributions of Random Walk models in Chapter 5

§ 2.7.5 Software Development

In order to test our constructs, it was necessary to implement the models and methods (algorithms) computationally. We chose to develop our methods and models in the form of design tools, to approach our ultimate goals. We chose a popular computational design platform (Rhino3D+Grasshopper3D) as a laboratory and testbed environment; and developed our tools using VB.NET and C#.NET languages, using Microsoft Visual Studio as an Integrated Development Environment (IDE).

Dot NET languages (VB.NET and C#.NET) ideally suited our project because they are Object-Oriented Programming (OOP) languages. OOP languages allowed us to invent constructs (e.g. Isovist Bubbles) and develop a library of methods operating on them. We have exploited this capacity in certain key areas, the most notable of which is the algorithm for enumerating plan-layout topologies; in which we created our own definition of topological edges, polygon meshes using n-sided topological polygonal faces and alike.

For the first two years of software development, VB.NET was used primarily. Later we switched all developments to C#.NET. Although both languages can achieve exactly the same things, the latter proved to be more suitable and much clearer for further developments. The syntax of C# is strict and clear as compared to VB.NET and thus prevents many mistakes and eases debugging.

Debugging is an inevitable part of any software development process. The matters we have been dealing with in our developments had very important spatial aspects, which could be best monitored visually. Choice of a relevant IDE is therefore based on its capabilities for showing results when developing code. It is not straightforward or practical to show spatial results (2D or 3D) in an IDE such as Visual Studio.

In practice, we adopted an agile approach in software development, involving rapid prototyping, testing, and restructuring. Therefore, we used Grasshopper3D and its Dot NET scripting components as our main IDE, where we could immediately test the implementation results. Grasshopper3D proved to be an appropriate lab environment for prototypical implementation and development of algorithms. However, in the end of any implementation & test cycle, we used Visual Studio for realizing sophisticated OOP constructs such as classes and dependencies. For developing the final version of our computational libraries, such as configraphics.dll, we used use Visual Studio as well.

§ 2.7.6 Verification and Validation

We have provided a variety of mathematical methods for analysing urban configurations, particularly in chapter 5. Here we clarify our stance as to how our methodology and its computational implementation (the toolkit as a software application) can be verified and validated. To avoid common misconceptions, we first recite the widely accepted meaning of the terms (from (Duncan, 1996) recognized by IEEE [Institute of Electrical and Electronics Engineers]):

- "Validation. The assurance that a product, service, or system meets the needs of the customer and other identified stakeholders. It often involves acceptance and suitability with **external** customers. Contrast with verification."
- "Verification. The evaluation of whether or not a product, service, or system complies with a regulation, requirement, specification, or imposed condition. It is often an **internal** process. Contrast with validation."

In other words, we can define these processes as to their guiding questions (Boehm, 1989):

- Validation: "Are we building the **right** product?"
- Verification: "Are we building the product **right**?"

As highlighted above, we believe a proper validation of a methodology such as ours should be done by **a disinterested third party** (which by definition does not have any ties or partnership with those offering similar methodologies). There is a trend to show quantitative analyses regarding correlation of centrality measures with actual movement patterns of people or distribution of such things as retail businesses, land-uses, or property values. We do not provide such studies for three reasons:

- Quantitative validation of our simulation models demands a separate research to be performed by a disinterested party;
- Our work is primarily on 'how to do measurements on walking and cycling accessibility potentials' not on predicting people's movements or activity patterns;
- We do not claim that our methods can predict actual movement patterns of people or their activity patterns, as we believe the real movement or activity patterns cannot be predicted merely by studying the network without taking into account land use or densities.

Here follows an account of what we have actually done for verification and validation:

Verification [internal feedback]: activities included ensuring the mathematical consistency of results, validity of geometric objects, and correctness of spatial distributions. Most of these were facilitated by proper visualization of results; however, mathematical consistency checks were performed on paper based on the formulations of the methods and models and their equations. Inputs and outputs have been inspected in terms of their consistency on a number of case studies (sample datasets) and extreme situations and/or extreme settings (e.g. a weight set to zero) have been checked.

Validation [external feedback]: activities included publishing of research outputs in the form of academic papers and revisions based on reviewers' feedbacks, releasing the main products as freeware applications to the professionals in the field and establishing two dedicated user groups as communication channels. In addition, external feedbacks have been collected during hands-on international workshops conducted with international colleagues.

Based on collected internal and external feedbacks we have revised the structure of our products several times; added new methods, which could be of interest; and in few occasions removed or substituted methods that were either unclear to the users or those which seemed to be of less practical use. An example of the major improvements (pertained to the work reported in Chapter 5) is the replacement of our former formulations of proximity and vicinity, respectively by Fuzzy formulations of accessibility Closeness-to-All and Closeness-to-Any.

§ 2.7.7 Crowd-Sourced Test & Validation

It was decided at an early stage of development that the main products of this project (i.e. the methodologies) should to be implemented, published, and tested continuously by their target users. The testbed platform Grasshopper3D provided the means to set up user groups, and so we established two open user groups as below:

- SYNTACIC (chapters 3 &4): *http://www.grasshopper3d.com/group/space-syntax*
- CONFIGURBANIST (chapters 5 & 6): *http://www.grasshopper3d.com/group/cheetah*

These groups have provided feedback on technical issues and usability of the methodologies. It is ideal to meet target users in workshops and engage in in-depth discussions; however, a crowd-sourced test and validation approach using dedicated social-professional networks such as the Grasshopper3D forum is complementarily effective. The GH platform was ideal for prototyping but we do not see it as an ultimate platform for deployment of our methodologies. Considering the current technological trends an ideal deployment platform should be web-based. As a strategy for future developments, we are considering Open Source publishing of our software products attached to scientific publications for ensuring maximum outreach and collecting feedbacks more effectively.

3 Model and Methodology A: Architectural Configuration

As mentioned in the dissertation outline in chapter 1, there are two parallel tracks of work reported in this dissertation: track A (on architectural configurations) and track B (on urban configurations). This chapter introduces a graph-theoretical design methodology for computational analysis and synthesis of architectural configurations as topological patterns. In addition, it provides 2D and 3D computational-geometry constructs for geometrically shaping spatial configurations. Simply put, the idea behind the work reported in this chapter was to devise a design methodology to allow for reaching a layout by gradually concretizing a functional-relational abstract graph[a].The chapter:

- Outlines the necessity and potentials of a configurational approach to architectural design;
- Provides definitions and examples of architectural configurations;
- Reintroduces network analysis measures for computational design;
- Explores the possibility of designing explicitly by means of topological configuration;
- Introduces 2D and 3D spatial smart agents (for Agent-Based Models aimed at geometrically shaping spatial configurations);
- Concludes by discussing how the number of design possibilities grows when designing through configuration; and
- Suggests ways of using configuration graphs and spatial agents for configurative spatial layout

Goal: To deliver a design methodology for spatial layout that brings spatial network analysis to architectural design process for assessing social, functional, or programmatic performance of a building configuration;

Question: How can we obtain an architectural layout from a spatial configuration graph, while controlling its performance? (Chapters 2, 3, 4)

a Some preliminary results of the research discussed in this chapter have been published before in two papers (Nourian, P. Rezvani, S., Sariyildiz, S., 2013) (Nourian, P. Rezvani, S., Sariyildiz, S., 2013)

§ 3.1 Background and Motivation

The functional importance of spatial configuration in architectural design is best understood in the context of complex buildings such as hospitals, airports, transportation hubs, museums, etc. In such buildings, there are often functional requirements as to what spaces should have a direct spatial link to what other spaces and sometimes on what spaces need to be disconnected from what other spaces. The reasons behind these requirements range from those pertained to safety and security, efficiency in movement of people, facilitating certain social or work-related interactions, and controlling, hindering, or blocking certain unwanted encounters and interferences. If not designed properly, problems in spatial configuration may cause disorganization, waste of precious time, safety issues, and even security breaches. The architectural motivation of this research was to propose an explicit, systematic, and direct way of addressing configurational issues in spatial layout, a process that could be equipped with real-time analytic feedback on the potential performance of a spatial configuration, in terms of complying with the intended functional programme. In order to provide this analytic feedback, we revisited some graph-theoretical centrality indices as performance indicators from theories in architectural morphology.

Space Syntax theory (Hillier, B., Hanson, J., 1984) & (Hillier, 2007) is an umbrella name for a body of methods and knowledge on spatial qualities of architecture as distinguished from its over-highlighted formal aspects. Ever since its inception, the focus of Space Syntax studies has shifted significantly from architectural scale to urban scale; however, the roots of the theories are clearly architectural. From an analytical point of view, Space Syntax theories and methods provide a framework for studying the effect of spatial arrangements on social interactions within space, human mobility, and accessibility. In other words, Space Syntax models provide for measuring what we can loosely term as 'social/programmatic performance' of buildings and built environments.

From another perspective, in a study of building types as social constructs, John Habraken (Habraken, 1988) categorizes three major aspects of building types as 'social constructs': spatial organization, physical structure, and stylistic systems. He suggests that the one most intimately related to our behaviour is the 'spatial organization'; he specifically mentions that a social role certain space has within a building is very much dependent on its 'position' as to the transition from public to private.

From a computational design perspective, the issue of 'plan layout' has been mostly addressed from various optimization point of views (Lobos, D., Donath, D., 2010); many of which deem configuration as an order that can be 'found' through thousands of trials and errors in putting spaces together in different ways in order to maximize certain qualities. This approach to plan layout is in deep contradiction to viewing architectural design as an intellectual activity initiated with 'proposing' configurative ideas. We believe "Architectural and urban design, both in their formal and spatial aspects, are seen as fundamentally configurational in that the way the parts are put together to form the whole is more important than any of the parts taken in isolation" (Hillier, 2007, p. 1). "Configuration as the way spaces are related to each other in order to serve a functional purpose is the very nature of architecture" (Hillier, 2007, p. 67); and yet we find very little about the way design can systematically proceed through dealing with such a matter.

What is primarily missing in the literature about computational layout is a methodological approach rooted in consideration of "how designers think" (Lawson, 1980); likewise, a practical design methodology for considering the social/functional implications of configurations is absent. Specifically, in the mentioned optimization approaches to plan layout, it is often neglected to relate to design processes as practiced by designers. Designers do not seek to reach an order through thoughtlessly trying out random arrangements of spaces; on the contrary, they usually start with an 'idea' as to how spaces should be put together to function in a certain desired way. Such configurative ideas convey the understanding of architects from the functional/programmatic requirements and/or what is 'socially' considered as desirable.

§ 3.2 Advantages of a Configurative Approach to Design

In this chapter, we propose a configurative approach to spatial layout. The prominent advantage of a configurational approach in design is to ensure reaching to a required set of spatial connection in designing -spatially or programmatically- complex buildings. In addition, configurational analysis can help in predicting the functioning of spatial configurations; especially where the spatial configuration is so complex and large that mere intuition cannot 'see' the so-called 'centrality' structures. Our proposed configurative (topological) approach to design (spatial layout) provides the possibility of using real-time feedback on the likely spatial functioning of proposed configuration, e.g. by using a spatial network analysis methodology such as Space Syntax.

It is intuitively obvious that in any spatial structure some locations are more central than others are. The centre-periphery spectra can be felt by common sense and seen to have influence on such things as rent prices, property values, and the location of retail businesses in cities and such things as accessibility and privacy and community spectra in buildings. An idea behind network analyses is that the structure of a network has strong influence over dynamics of phenomena such as popularity of places by virtue of human movement, spread of information, and diffusion processes on networks.

One [indirect] way of studying dynamics in complex networks is through studying the network structure by means of measures that can reveal the heterogeneity of nodes such that we can rank them as to their 'structural importance' or 'centrality'.

The architectural or spatial relevance of such analyses is that they could reveal why certain places tend to be more communal or public whereas some others tend to be more private. In other words, the match between patterns of social interaction and spatial structure is an inherently architectural question.

A generic question is how a spatial configuration should be arranged[25] so as to encourage and facilitate certain types of social encounter while discouraging or hindering certain other types of encounter. Our proposed approach does not directly answer such questions but facilitates the intellectual reflection by designer on such performance issues by means of providing configuration graph visualizations and network centrality indicators. It must be noted that studying centrality should not be mistaken by a search for the 'best nodes'. The best location for a pub is not necessarily the best location for a school or an apartment -let alone the different preferences of people for their residence location in a city.

What elevates 'architecture' from 'building design' has been prominently associated with the aesthetical aspects of architectural profession, perhaps because of their concrete manifestation. Another perspective can be explored from a social-scientific point of view, from which architecture can be seen as 'the art of providing avoidance and encounter where necessary, in order to facilitate desired social interactions and hinder unwanted ones'.

Speaking of the social, we can observe that a social-scientific stance is closely related to modelling the aggregate behaviour of individuals and thus it is related to a psychological analysis of individual behaviour in space. This is the subject area of environmental psychology, best exemplified in the work of the pioneer J.J. Gibson, namely: (Gibson, 1977) (Gibson, 1979). Note that in configurational analysis we do not seek to model or predict the behaviour of an individual in space, for such an effort would be philosophically futile[a], instead, we are interested in the social behaviour or the statistically aggregate behaviour of individuals. In this work, however, we merely provide a basic set of tools for modelling spatial networks that can be used as bases for such models, but not the statistical models of the actual spatial behaviours. Behavioural modelling as such falls outside the scope of this project.

a For reasons of anti-causality mentioned in the chapter 2, we cannot model human behaviour as cause-effect relations, therefore we cannot predict individual behaviour

§ 3.3 Definition of Architectural Configuration

In light of the abovementioned clarifications, we define architectural configuration as a graph composed of nodes representing individual spaces and links representing immediate spatial connections such as doors in between rooms. In other words, this representation comes close to what architects conventionally draw as bubble diagrams. Before giving a formal definition let us see some examples from architectural practice.

§ 3.3.1 Bubble Diagrams

As apparent in the example shown in Figure 6 and Figure 7, bubble diagrams are conventionally used to reflect upon and/or design spatial arrangements prior to designing spaces as concrete geometric shapes. Architects, however, do not draw such diagrams according to a set of clearly defined rules.

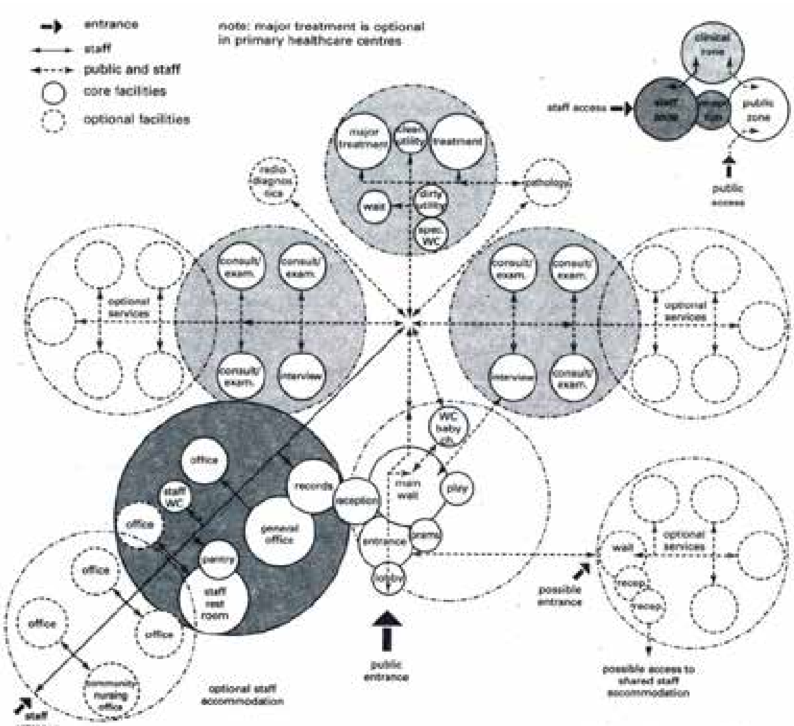


FIGURE 6 a bubble diagram with links as directed edges indicating corridors in a hospital, from Neufert Architect's Data (Neufert, Ernst, Peter Neufert, and Johannes Kister., 2012)

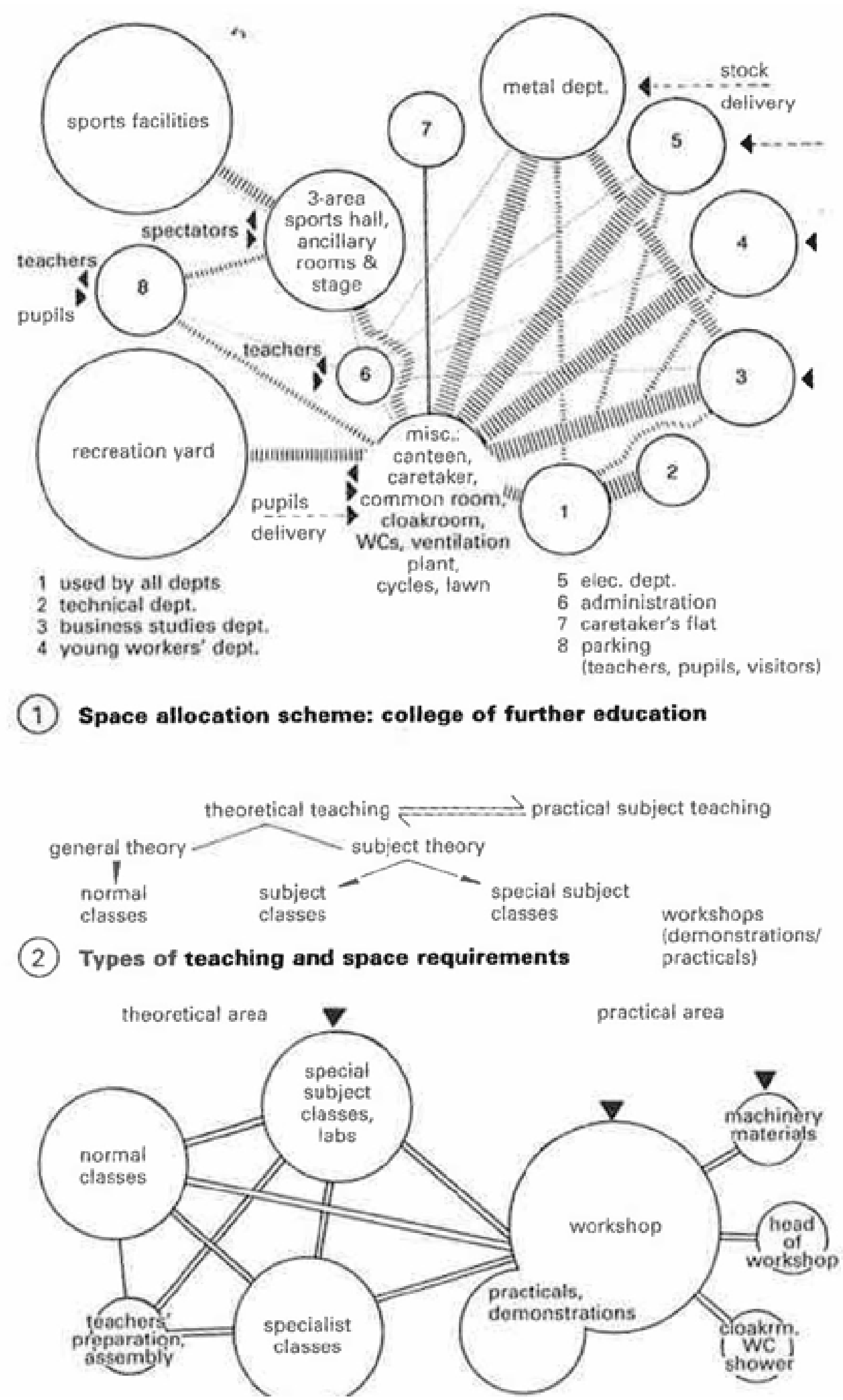


FIGURE 7 an exemplary bubble diagram representing the spatial relations for a proposed technical from Neufert Architect's Data (Neufert, Ernst, Peter Neufert, and Johannes Kister., 2012)

§ 3.3.2 Mathematical Definition

Formally, we can conceptualize bubble diagrams as labelled graphs whose nodes can have attributes such as area properties or alike. In our proposed methodology, we have technically defined them as graphs (represented as either adjacency lists or adjacency matrices) with associated lists of attributes for nodes. Formally, we can define an architectural **configuration graph** denoted as $\Gamma = (\mathrm{N}, \mathrm{L})$, i.e. the graph Gamma as an ordered pair of nodes and links. The set N includes the indices of the nodes, each of which representing a space and the set L represents the set of connections between nodes that is practically a set of node tuples. This means that the set of links is a subset of $N \times N$, in which the operator $\times$ denotes Cartesian product of the two sets. In other words, this is denoted formally as $L \subset N \times N$. We shall clarify our choice of terms later in explaining § 3.5 preliminaries.

There is one inherent -and rather delicate- limitation with representing architectural configurations as bubble diagrams in representing corridors. The point is that corridors and the whole circulation system of a building might not be initially thought of as spaces necessary for the design, as they are only to make the building operational, but they are not directly sellable spaces. It is therefore not straightforward to conceive of them in an early phase of design process when tackling a list of functional requirements. As can be seen in Figure 6, architects sometimes try to explicate such corridors when they are critically important, however in a rather inaccurate way. In our work, we decided to work on a simplified case where the connections only indicate immediate spatial adjacencies such as doors, acknowledging this limitation. However, if the designer is certain and clear about the necessity of existence of a corridor or hallway as a space to be designed, it can also be treated as another space just like others. Ideally, though, it would have been best to propose a tool for explicitly designing a circulation system as well. This would be a too specific task, which would fall out of the scope of this work.

§ 3.3.3 REL charts and From-To Charts

A clearer representation of architectural configuration, which is also known to architects, is an adjacency table (similar to a matrix) as shown in Figure 9 and Figure 11. Such adjacency tables are technically referred to as REL charts (Activity Relationships Charts) or From-To Charts and they are commonly used in Facilities Planning (Tompkins, J. A., White, J. A., Bozer, Y. A., 2010), which is an area of work in Industrial Engineering. If simplified by considering a connection wherever thought to be necessary, an example table as such can be considered as an adjacency matrix that is to be realized as a spatial configuration or plan layout. For reasons of practicality, it might turn out necessary to realize many of the suggested adjacencies as corridors while realizing the rest as doors. As obvious, such a table does not indicate whether the connection should be a door, a corridor, or a staircase. The specifications in such tables can be given as *connection: mandatory*, *connection: desirable*, and *connection: neutral, or connection: undesirable*; or as is more common in Facilities Planning as below:

- **A**: Absolutely Necessary
- **E**: Especially Important
- **I**: Important
- **O**: Ordinary
- **U**: Unimportant
- **X**: Undesirable

Such degrees of connectivity might be freely interpreted in the design process as doors, corridors, and possibly staircases. The important point is that such a table is driven purely by functional necessities rather than formal considerations; but once explicated as such it finds some graph-theoretical and topologic connotations that we shall try to bring closer to geometric possibilities.

In REL charts as shown in Figure 8, Figure 11, the relationships are considered to be undirected. If directed relationships are to be represented then a From-To chart as shown in Figure 9, Figure 12, and Figure 13 is used.

In this work, for simplicity, we have not used directed From-To charts or even REL charts for putting in activity relationships; and used (Point-Line) bubble diagrams as inputs for reading desired relationships between spaces (see Figure 19).

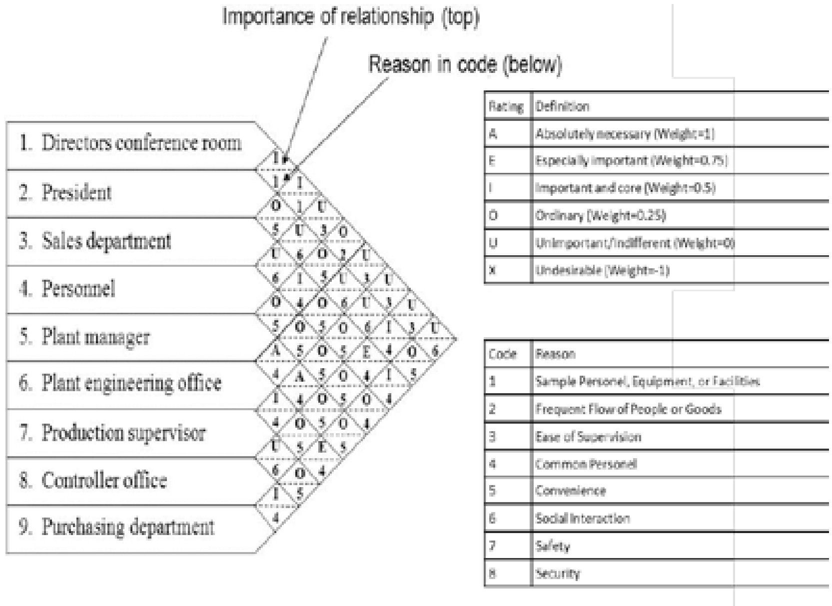


FIGURE 8 a REL chart example Image reproduced from (Tompkins, J. A., White, J. A., Bozer, Y. A., 2010)

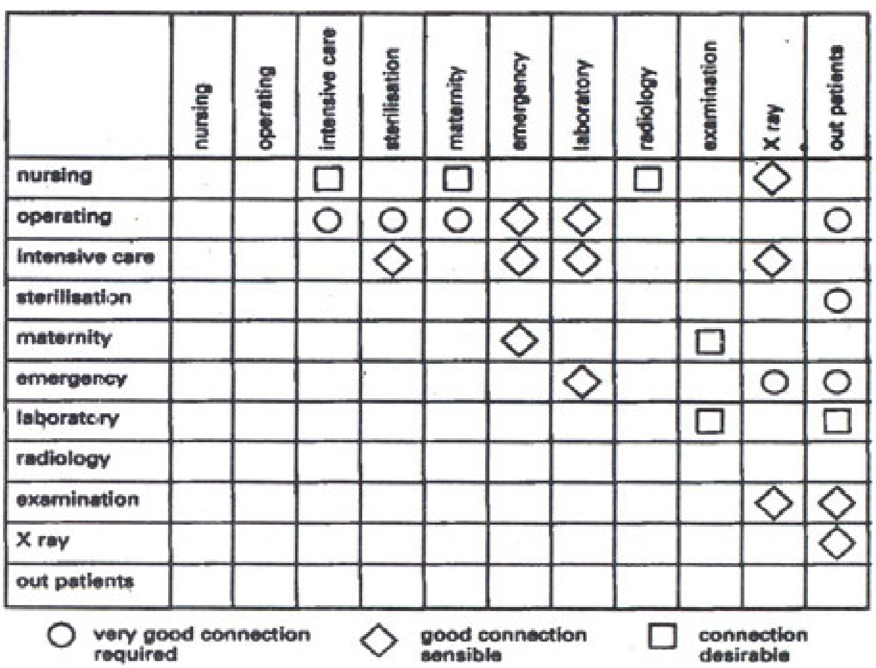


FIGURE 9 an adjacency table (matrix) prepared by architects to explicate the required spatial links among rooms and sections in a hospital, from Neufert Architect's Data (ibid). This table can be simplified into an 'adjacency matrix' representation of an architectural configuration graph.

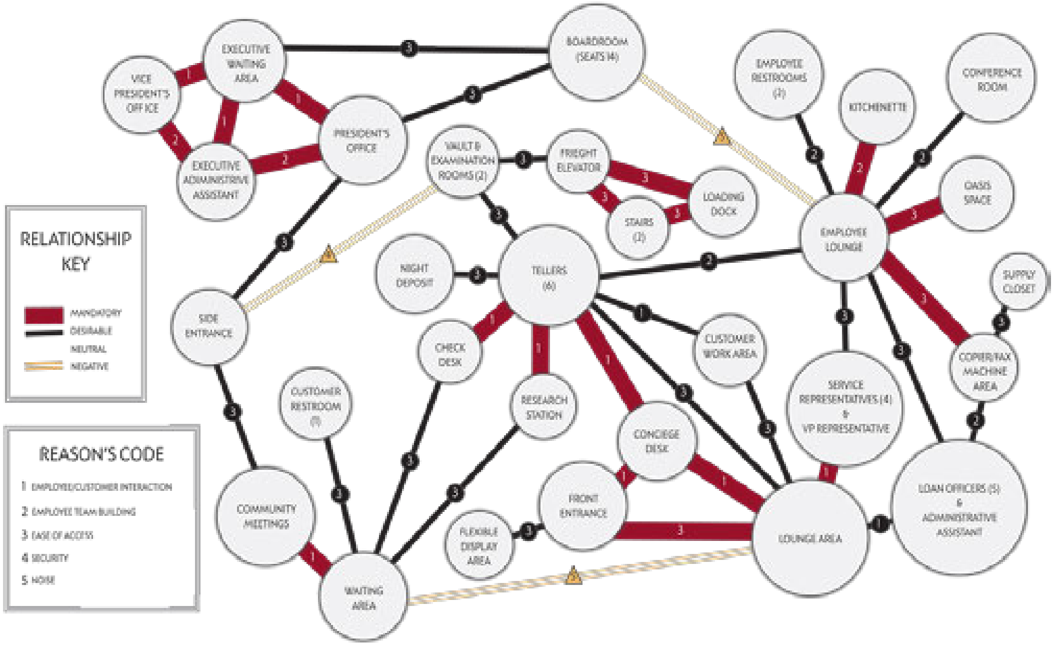


FIGURE 10 (Image courtesy of Kate Killebrew from *http://www.coroflot.com/killebrew/florida-bank*) shows a bubble diagram drawn based on the adjacency table shown in Figure 11. For the other diagrams and plans designed based on these diagrams see the website mentioned above.

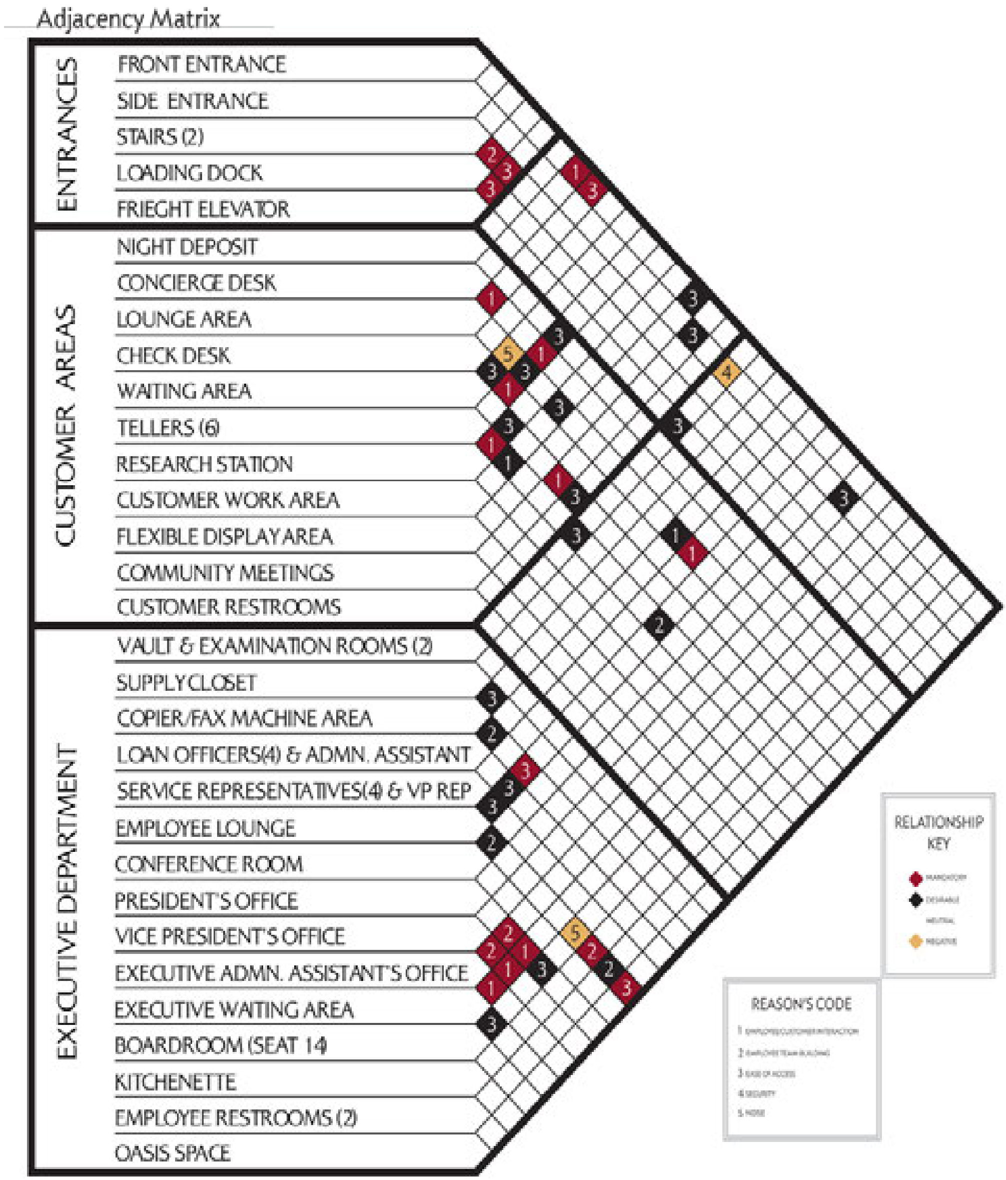


FIGURE 11 (Image courtesy of Kate Killebrew from *http://www.coroflot.com/killebrew/florida-bank*) shows an adjacency matrix that is drawn in a triangular format. Note that as adjacencies are undirected. This triangular table corresponds to a -45° rotated-half of an adjacency matrix that is symmetric. Relationship Key: Red=Mandatory, Black=Desired, and Yellow=Negative.

	Departments								
Departments	1	2	3	4	5	6	7	8	9
1. Reception		U	E	O	U	U	U	A	O
2. Emergenccy Unit			I	U	A	I	U	U	U
3. Outpatients Clinic				U	U	O	U	U	E
4. Wards					U	I	O	U	O
5. Intensive Care						E	I	U	O
6. Surgery							U	U	I
7. Laboratory								U	E
8. Administration									O
9. Farmacy									

Rating	Definition
A	Absolutely necessary (Weight=1)
E	Especially important (Weight=0.75)
I	Important and core (Weight=0.5)
O	Ordinary (Weight=0.25)
U	Unimportant/Indifferent (Weight=0)
X	Undesirable (Weight=-1)

FIGURE 12 an exemplary From-To chart which represents undirected connections, image reproduced from the educational materials of the course E212: Facilities Planning and Design at the Republic Polytechnic

	Departments								
Departments	1	2	3	4	5	6	7	8	9
1. Reception		U	E	O	U	U	U	A	O
2. Emergency Unit	U		I	U	A	I	U	U	U
3. Outpatients Clinic	E	I		U	U	O	U	U	E
4. Wards	O	U	U		U	I	O	U	O
5. Intensive Care	U	A	U	U		E	I	U	O
6. Surgery	U	I	O	I	E		U	U	I
7. Laboratory	U	U	U	O	I	U		U	E
8. Administration	A	U	U	U	X	X	X		O
9. Farmacy	O	U	E	O	O	I	E	O	

Rating	Definition
A	Absolutely necessary (Weight=1)
E	Especially important (Weight=0.75)
I	Important and core (Weight=0.5)
O	Ordinary (Weight=0.25)
U	Unimportant/Indifferent (Weight=0)
X	Undesirable (Weight=-1)

FIGURE 13 an exemplary From-To chart which represents directed connections, image reproduced from the educational materials of the course E212: Facilities Planning and Design at the Republic Polytechnic

§ 3.4 Architectural Spatial Network Modelled as a Graph

When referring to spatial connections, we apply the convention that reserves the term adjacency for referring to spatial links between features of the same dimension and the term incidence for the intersections between features of different dimensions. We shall see in chapter 5 how an adjacency matrix can be derived from an incidence matrix in case of a constellation of points and lines.

For modelling architectural configurations, we represent spaces (e.g. as convex or star-convex) as nodes and the immediate spatial connections between them as links. From this definition, it is apparent that if we take a simplified plan drawing into account and model its cells as nodes and adjacencies between cells as links then the configuration graph that we are interested in is a subset of this adjacency graph. An example of such configuration-graphs is drawn in Figure 14.

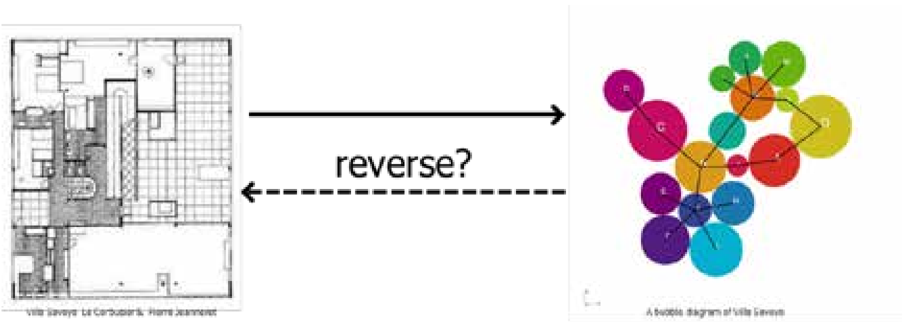


FIGURE 14 a spatial network representation of 2nd level of Villa Savoye (Le Corbusier & Pierre Jeanneret 1928) the connectivity graph is extracted manually, but the drawing is done automatically.

§ 3.5 Preliminaries of Modelling Spatial Networks

In this section, we clarify a number of notions essential to this work, namely our terminology, the notion of a Simplex, the concept of Poincare duality, and the difference of Graph Theory and Topology.

It is common to associate graphs with graph drawings immediately, perhaps because of the history of graph theory and the famous riddle of bridges of Konigsberg formulated and solved by Leonhard Euler (1707-1783). Although having a geometric intuition of a graph as a drawing made up of points or lines, respectively representing nodes and links, might be insightful; it is not necessary at all for a graph to be visualized or drawn to exist or to be dealt with. In fact, such a connotation might lead to basic confusions in dealing with the kind of configuration graphs that we introduce in the next chapters. To break the mental association between a graph and its drawing, we find it useful not to use the more common terminology (vertices and edges) when speaking of graphs. For a clear terminology, we shall use different terminologies in different contexts as shown in the table below:

N-D FEATURES	GRAPH THEORY	TOPOLOGY	GEOMETRY
0D	Node	Vertex	Point
1D	Link	Edge	Line (Curve)
2D	Cycle*	Face	Polygon (Surface)
3D	Clique*	Body	Polyhedron (Volume)

TABLE 2 Our terminology for n-dimensional primitives (features[26]) in different application contexts. * We figuratively use these terms to refer to 3-vertex and 4-vertex cliques in graph theory for the sake of completeness of the terminology; however, we do not use them in this work. In this table and the following schemas, we emblematically use a cold to warm colour scheme to indicate the spectrum between abstract functional requirements and concrete forms.

In dealing with topological[27] entities (i.e. in defining vertices, edges, faces, and bodies), embedded in 3D space, it can be advantageous to work with simplex primitives because of their nice properties such as ensured convexity:

- 0-simplex: Point
- 1-simplex: Line
- 2-simplex: Triangle
- 3-simplex: Tetrahedron

A k-simplex is mathematically defined as a k-dimensional polytope that is the convex hull of its $k + 1$ vertices. This means that any point within the space defined by the simplex can be represented by a linear interpolation of the simplex vertices. Formally, these vertices are defined as $v_0, \dots v_k \in \mathbb{R}^n$; and they are linearly independent, such that no vertex can be defined by linear interpolation of the rest of vertices (i.e. all corner vertices are absolutely necessary to define the space defined by the feature properly). The simplex cell C is then defined as the locus of points defined by linear interpolations of the corner vertices $v_i$with barycentric coordinates denoted by α_i, as follows:

$$C = \{\alpha_0 v_0 + \cdots + \alpha_k v_k | \alpha_i \geq 0, 0 \leq i \leq k, \sum_{i=0}^{k} \alpha_i = 1\} \quad (1)$$

The apparently obvious definition of a configuration graph becomes rather problematic or challenging when analysing built space. This is because arriving at such a spatial decomposition is non-trivial and often problematic, especially if it is to be done automatically on maps. The common notion is that the entirety of space has to be discretised by decomposing it to convex units (e.g. simplices, pixels or voxels or convex polygons/polytopes) such that these convex units can represent the nodes of a spatial network whose links indicate adjacency (immediate spatial connection) among such cells. Following this notion, and provided a tessellation of space in terms of convex cells, we can represent a 'spatial network' as a 'dual graph'.

Based on the Poincare Duality theorem (similar to the approach of (Pigot, 1991) and (Lee, 2001)), we establish a pairing between k-dimensional features and dual features of dimension n-k, where n denotes the dimension of the space within which the features are embedded. Such dualities are shown in tables below:

PRIMAL	DUAL
0D vertex (e.g. a point)	1D edge
1D edge (e.g. a line segment)	0D vertex

TABLE 3 Duality of features in 1D space

PRIMAL	DUAL
0D vertex (e.g. a point)	2D face
1D edge (e.g. a line segment)	1D edge
2D face (e.g. a triangle or a pixel)	0D vertex

TABLE 4 Duality of features in 2D space

A well-known example of duality between 2D maps is the duality between a Delaunay triangulation and a Voronoi tessellation.

PRIMAL	DUAL
0D vertex (e.g. a point)	3D body
1D edge (e.g. a line segment)	2D face
2D face (e.g. a triangle or a pixel)	1D edge
3D body (e.g. a tetrahedron or a voxel)	0D vertex

TABLE 5 Duality of features in 3D space

An example application of dual relationships in 3D is shown in Figure 15. A face in the left image can be considered as the element through which two 3D cells are connected; this is why representing the same face with an edge in the dual graph makes sense as it connects two vertices representing the respective 3D cells. Considering an edge in the left image, we can observe that an edge is incident to four 3D cells, i.e. its dual face is incident to four respective dual vertices. Similarly, a vertex in the left image is incident to eight 3D cells, which is associated with a dual body in the right image composed of eight corner vertices corresponding to the respective cells in the left image.

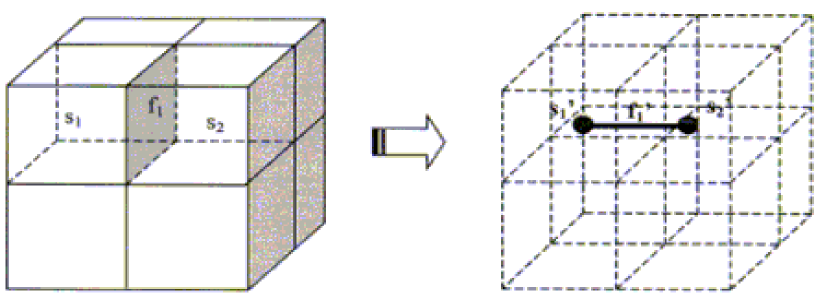


FIGURE 15 representing adjacencies between 3D cells or bodies via their dual vertices (Lee, 2001)

When referring to spatial connections, we apply the convention that reserves the term adjacency for referring to spatial links between features of the same dimension and the term incidence for the intersections between features of different dimensions.

To avoid common confusions about the distinction of topology and graph theory, let us clarify our terminology. We call a graph a virtual object consisting of a set of entities considered as nodes (virtually considered as vertices) and a set of associations (of any kind, but the same for all associations) as links (virtually considered as edges). A graph needs not to have a geometric representation to exist. This is why we prefer to refer to elements as nodes and links rather than vertices and edges before establishing a graph drawing (as the latter terms have a stronger geometric connotation).

The study of the ways a graph can be drawn (geometrically represented), is technically referred to as embedding; and is a subject of topology and in particular topological graph theory. Relating to the conventions of topological graph theory and topology, we denote a topological graph G as a set of vertices (nodes) denoted as V and edges (links) denoted as E. This is written as below:

$$G = (V, E) \tag{2}$$

If an embedding (a drawing of the graph on a kind of topological surface such as plane or sphere) is known, then there is a new list of objects to be mentioned: these are faces (loops) denoted as F. A face of an embedding is a component in the defined decomposition of the plane. Mathematically, the faces are defined by their boundaries, i.e. usually an ordered set of edges (or vertices).

A graph with a topological component F is called a Map. Interesting enough, there is also a class of maps used in computer graphics and CAD applications called Polygon Mesh, i.e. a [triangulated] map. Meshes are also represented and stored as three sets of objects as below[28]:

$$M = (V, E, F) \tag{3}$$

Euler formula (Euler characteristic of convex polyhedrons) relates the number of vertices to the number of edges and the number of faces for an embedding/planar drawing. Euler Relation for Planar Graphs: Every embedding (map) of a connected graph in the plane (or a surface) satisfies equation (4) a.k.a. Euler-Poincare Characteristic[29] for a map (mesh) M: (Edelsbrunner, H., & Harer, J., 2008, p. p.22).

$$\chi(M) = |V| - |E| + |F| = 1 \tag{4}$$

As an example, we show three different mappings/embeddings of a single graph the figure below. Note that the three entities are the same from the graph theoretical point of view but they are topologically different.

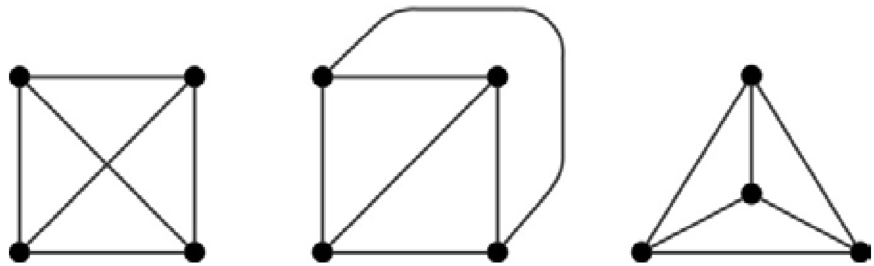


FIGURE 16 three drawings of K4 (the complete graph of order 4). From left to right: a drawing that is not a valid [2-manifold] embedding, and embedding with one curved edge, and a straight-line embedding (Edelsbrunner, H., & Harer, J., 2008).

§ 3.6 Configurative Design Methodology

The centre of this methodology is the bubble diagram that the designer-user provides. We find bubble diagrams of great interest from different viewpoints. Firstly, bubble diagrams have been known to architects for a long time as a way to think of spatial arrangement in plans in a rather methodical fashion. Second, a bubble diagram can be interpreted as a graph of spatial connectivity and/or adjacency; this provides for configurational analysis to be done at the early phases of a design process. Third, if drawn as a planar graph (a graph drawing without crossings on a plane sheet) it can be considered as a map, which has another component in addition to the aforementioned graph that makes it much more interesting for studying plans: topology. In short, our investigation starts from Graph Theory, continues with Topological Graph Theory (concerning the issue of embedding graphs in Euclidean space), goes into Computational Graph Drawing and enumeration of distinct possibilities using Combinatorics, and then ends in the domain of Computational Geometry. This process is schematized in Figure 17.

Our design methodology is a fusion of what was proposed by Steadman (Steadman, 1983, pp. 69-75), Tutte's (Tutte, 1963) convex drawing algorithm, a force directed graph drawing algorithm, several minor algorithms, real-time Space syntax analyses, and two computational geometry constructs (Isovist Bubbles) that can be regarded as smart spatial agents. Following our design methodology, the designer is free to insert a configurative idea and change it as they think is best, both at the beginning and during the process[a]. In a manner of speaking, our methods are meant to reveal meanings and implications of such configurational inputs.

a Our process does not automate the design process in any sense.

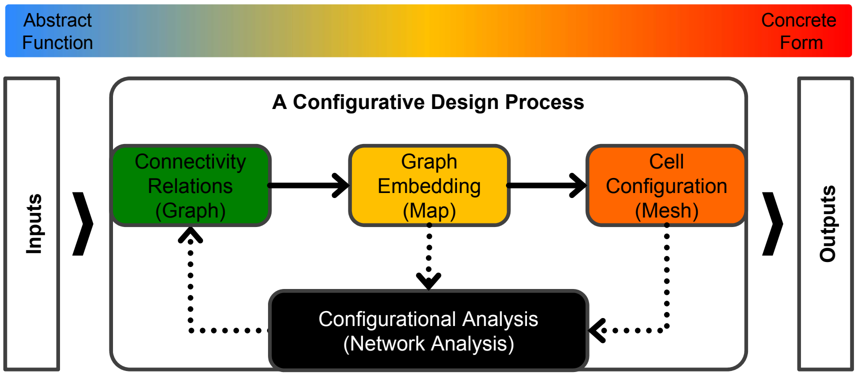


FIGURE 17 the SYNTACTIC design methodology described in terms of the gradual evolution of its representations from abstract 'graphs' to more tangible topological 'maps' to concrete cell configurations as polygonal meshes with a correct geometry

The advantage of using Space Syntax in analysing bubble diagram sounds apparent, but its application has been rather neglected. On the other hand, there have been quite a few rigorous investigations on generation of plan layouts, some of which have even started with bubble diagrams (a recent survey of such methods can be found in (Lobos, D., Donath, D., 2010)).

Our focus was on what is theoretically possible to be done in a design process following this path, which was to start with a graph and end up with a plan. We will see how a graph can be gradually concretized towards a geometric spatial configuration through topological layouts. By inserting assumptions about the possibilities, such as confining the layout process to rectangular shapes we could possibly end this process with geometric layouts. However, regarding generality and usability of outputs, we decided to go away from this process in favour of a novel approach based on computational geometry objects called Isovist Bubbles (§ 3.12 & § 3.13).

We have implemented this design methodology in a computational design toolkit[30] named SYNTACTIC, a.k.a. Space Syntax for Generative Design. The workflow that this toolkit puts forward is explained in the next chapter. In the followings, we explain the basic steps essential for the inception of a syntactic layout process in our methodology. This design methodology is explained below:

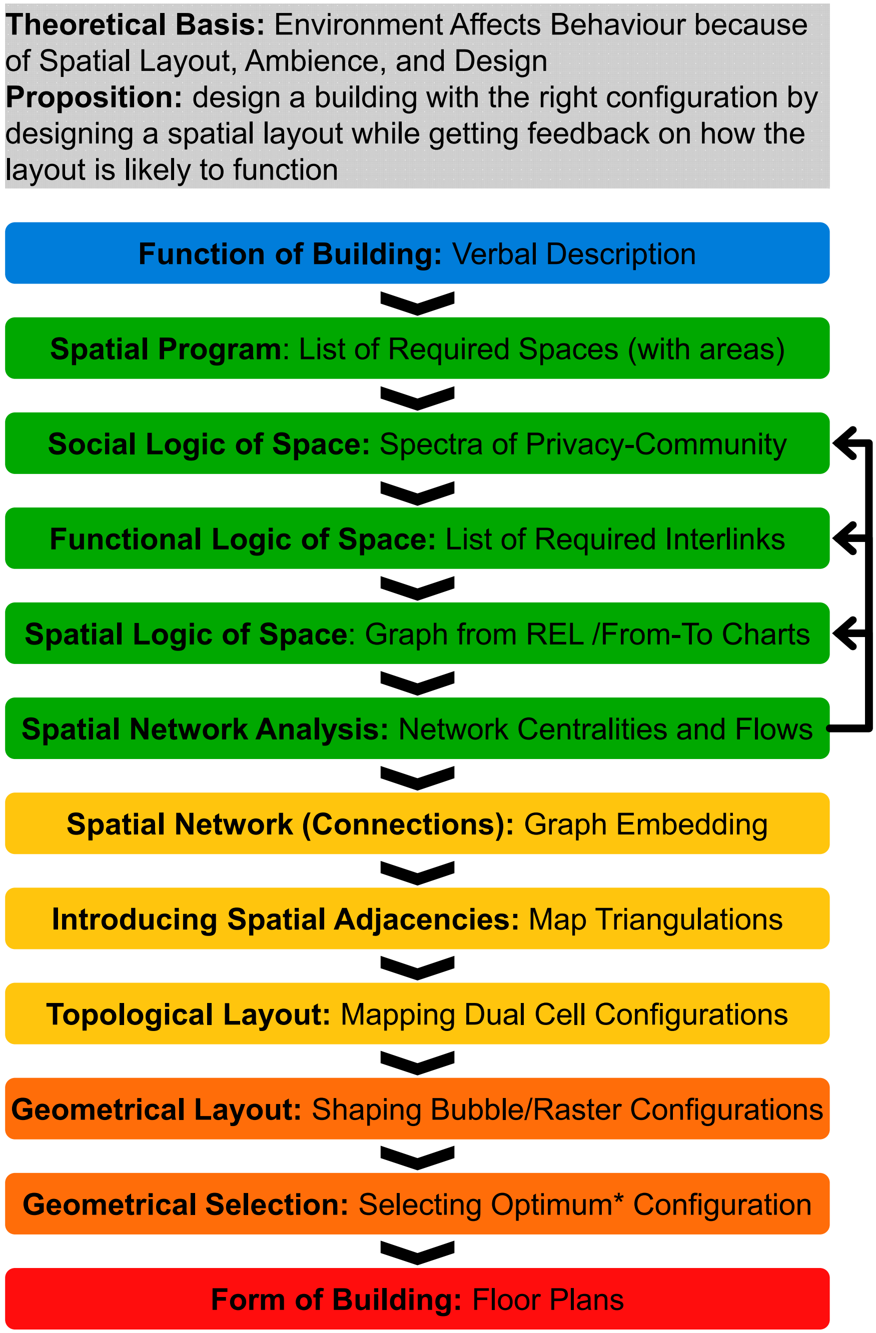


FIGURE 18 the workflow put forward by our syntactic design methodology

§ 3.7 Configurational Analysis vs Configurational Synthesis

Methodical interest in spatial configurations seems to have been originated or at least flourished in two lineages of work best exemplified in/traced to the books "The Social Logic of Space" (Hillier, B., Hanson, J., 1984) and "Architectural Morphology" (Steadman, 1983) and "The Geometry of Environment" (March, L, Steadman, P, 1974). The first book follows an analytic approach to configuration graphs whereas the latter ones explore the idea of synthesis of form from configuration graphs. Bill Hillier and Julienne Hanson were pioneers of configurational analysis (analysing spatial configurations in terms of their effect on human-space interactions), while Philip Steadman and Lionel March were the pioneers of configurational synthesis (producing geometrical/spatial layouts based on configuration graphs). The main contribution of this chapter can be seen as connecting these two lines of work by proposing a computational design methodology. Before going into depth of the methods, we need to begin by giving an accurate definition of architectural configuration.

We need to distinguish two cases in defining architectural configuration: when 'analysing' architectural configurations, we might be dealing with a concrete geometry of environment, whereas in 'designing' architectural configurations, like the case of plan layout in 2D, we might only have an abstract plan and a few geometric constraints. This distinction is essential in defining what an architectural configuration is. Technically, we could say an architectural configuration -in its most abstract form- is a graph, and while being processed further in a design process, it can be embedded topologically in 2D, or 3D space, i.e. as a topological map; and finally when the spaces (rooms) are geometrically shaped it has additional geometric properties. When it is a finalized geometrical layout, it can be analysed in terms of such geometric properties as 'visibility'. For these reasons, in our 'configurative' design methodology, we focus on defining architectural configuration as a graph that defines spatial arrangement or the way the spaces in a building are supposed to connect to one another. We shall also address the visibility properties of spatial nodes (e.g. rooms) at the end of this chapter.

§ 3.8 Configurative Design Process

In this chapter, we present a computational design methodology for architectural layout that combines configurational analysis with configurational synthesis. The aim of this methodology is to provide for a systematic approach to design through which the spatial performance of a design is reflectively shaped by providing the designer with real-time feedback on social/spatial implications of a layout. The methodology is called SYNTACTIC[a] design, emphasizing the role of syntactic or configurational analyses in the process. Both analytic methods and synthetic methods of the methodology make use of a labelled graph that is designed by (and shown to) the user as a 'bubble diagram'. The core idea is to build upon the intuitive understanding of a designer from a bubble diagram and use it as a medium for conveying configurative ideas to analytic and synthetic algorithms. In other words, the proposed design methodology was to give life to the bubble diagrams conventionally used for spatial arrangement, to allow for communication of configurative ideas between designers and computers. This idea brought about the following technical questions:

- What spatial qualities possibly result from the proposed connectivity patterns; and how can we study them methodically? (Spatial Network Analysis, e.g. Space Syntax)
- How can a computational system interpret configurative ideas -put in the form of a bubble diagram- to plan layout patterns? (Using topological modelling methods)
- Does a certain configurative diagram have only a single corresponding layout or more; if there are many, how can we systematically find the fundamentally distinct ones? (Considering shape constraints, they can be enumerated topologically)
- How can a topological layout be concretized into a geometrical layout? (Computational Geometry constructs & geometric constraints)

As a response to the above questions, we put forward the following configurational design process, which:

- begins with an abstract configurative arrangement of spatial entities (connectivity graph) proposed by a human designer;
- provides an interactive representation of the above graph as a bubble diagram;
- resumes by enumerating all possible topological interpretations of the configurative inputs as plan layout patterns; and
- can eventually end with geometric specification of the layout according to the design brief; we currently introduce computational geometry agents for forming the individual spaces; the agent-based model to form the whole layout is under development.

a This is the name of a toolkit that embodies our implementation of this methodology,; the toolkit is introduced in the next chapter.

A single connectivity graph, as an abstract entity, is interpretable to various geometric configurations, all of which share the same pattern of spatial interconnectivity although they may vary in size and shape from one another. Using our methodology, designers can sketch how the spaces are to interlink, and then they can use our methodology as follows:

- a set of methods interpret these interlinks as a graph that captures the spatial configuration (spatial connectivity structure) of a building;
- an algorithm draws a neat drawing of the architectural configuration graph as a bubble diagram;
- another set of methods assist the designer to find a catalogue of all possible planar topological embedding of this graph;
- a set of methods provide instant feedback on configurational qualities using graph theoretical measures of Space Syntax such as depth (visualized in justified graphs), integration, control, entropy, choice, and difference factor (Hillier, B., Hanson, J., 1984) & (Hanson, 1998).

two computational geometry agents (2D & 3D smart/goal-seeker objects) help the designer set up an agent-based model in order to geometrically shape the layout, based on the connectivity graph already constructed.

§ 3.8.1 Reading a Configuration Graph

The most critical step in the process is the inception phase where we need to interpret configurational ideas of the designer as a graph (e.g. from a point-line drawing as in Figure 19). Although this sounds to be a trivial task, in fact it is not. If we were to do this on a piece of paper, then we could ask the user to put some dots on the paper and mark them as rooms or spaces, then specify the connections between them. This means we need to take links after nodes; but before that, we need to construct a topological model in which there are no duplicate lines or points. This can be regarded as an implementation detail; therefore, we discuss it further in the next chapter. The point is, the graph that we need to form should describe adjacencies between nodes, or 0D features, whereas we receive points (0D) and lines (1D) as inputs. The detailed points regarding the user interface will be further discussed in the next chapter, which is on our implementation of SYNTACTIC.

Algorithm 1 explains the (simplified) steps required to translate a point-line drawing to an architectural configuration graph, both as an 'adjacency lists' and an 'adjacency matrix' representation. The last step (Case 2) and its equation in this algorithm are explained in chapter 5.5 (constructing spatial network graphs).

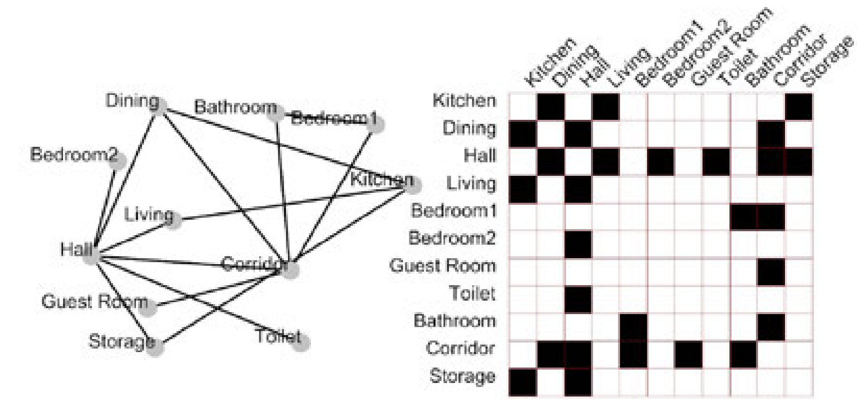


FIGURE 19 (left) an exemplary set of labelled points and lines to be interpreted as an architectural configuration graph; (right) a matrix plot representation of the inferred configuration graph.

Algorithm 1: Reading a configuration graph[a]

Given a point-line drawing of a graph by user

Output the graph $G = (V, E)$ as adjacency lists and / or adjacency matrices

- ❖ ensure there are no duplicates in the list of points;
- ❖ index the unique points as a list of vertex-points;
- ❖ match a list of input tags and a list of input attributes with the vertices;
- ❖ ensure there are no duplicates in the list of lines;
- ❖ index the unique lines as a list of edge-lines;
- ❖ find incidences among edges and vertices form adjacency lists & an adjacency matrix;
- ❖ choose between an adjacency matrix or adjacency lists:
  - ➢ case 1 (adjacency lists required):
    - ▪ form Connections as an N-dimensional array of lists;
    - ▪ for each edge-line in edge-lines:
- ❖ if both edge points are in vertex-points;
  - ➢ find index of edge-start SI;
  - ➢ find index of edge-end EI;
  - ➢ if Not Connections[SI]∋EI then;
    - ▪ Connections[SI].Add(EI);
  - ➢ if Not Connections[EI]∋SI then;
    - ▪ Connections[EI].Add(SI);
    - ▪ return Connections;
  - ➢ case 2 (an adjacency matrix required)
    - ▪ fill in a Vertex-Edge incidence matrix VE;
    - ▪ transpose VE and store it as EV
    - ▪ return the adjacency matrix $\boldsymbol{VV} = \boldsymbol{VE} \times \boldsymbol{EV}\text{-}\boldsymbol{D}$;

a After Michael Batty, a New Theory of Space Syntax (Batty, 2004), a more detailed account of this process is explained in § 5.4

§ 3.8.2 Drawing Bubble Diagrams

In order to free the designer from geometric constraints at the beginning of a design process, we offer a 'force-directed' graph-drawing algorithm (similar to (Eades, 1984)) (Algorithm 2) to draw a bubble diagram (see examples in Figure 21) merely based on nodes, links, and the intended area for the nodes. This way, the designer does not need to manage to draw a neat diagram, as the system does it for them; instead, they can think of what spatial connections are needed because of what spatial qualities.

In other words, the nodes can be anywhere in the space and if the links cross each other it does not matter; because the graph drawing algorithm will make a neat diagram out of this set of nodes and links [graph] if possible. That is, if the graph is a planar graph and that the spaces surrounded by other spaces could fit in their middle. A planar graph is a graph that can be drawn on a piece of paper such that none of its edges would cross each other. Note that even if the graph is planar, there are cases where drawing a coin graph drawing (a.k.a. kissing disk drawing, as produced by this algorithm) would not be possible simply because a coin might be too big to fit within its neighbour coins!

The force-directed graph-drawing algorithm is one of the most popular graph drawing algorithms but it is not guaranteed to work under all circumstances. There are no proofs of its convergence in the literature and hence we do not claim that it would work even if the graph is planar and the disks could fit nicely next to each other. Despite this theoretical pessimism, by tweaking the knobs of this algorithm, relatively nice[a] drawings can be achieved in very short time for complex graphs.

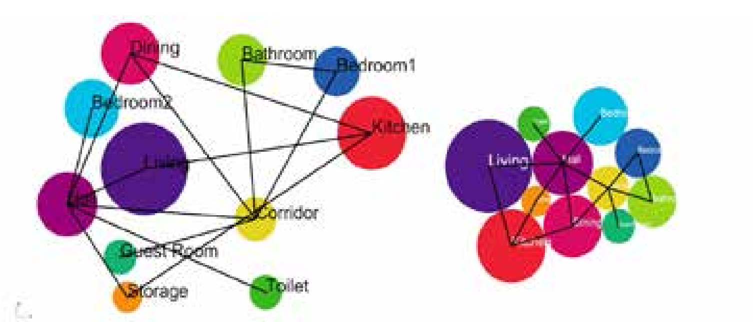


FIGURE 20 the left image shows a bubble diagram from the user with disks sized as to the desired areas; the right image shows the neat bubble diagram drawn by our algorithm.

a Nicety refers to a concept in the Graph Drawing literature that regards the measureable aesthetics of graph drawings in terms of such things as maximum distinction between nodes, avoidance of link crossings, etc.; all of which would help a human to understand a (usually large) graph.

Algorithm 2 Force-Directed (Kissing-Disk/Coin-Graph) Drawing

Given the graph $\Gamma = (V, E)$, $E = (V_i, V_j)$ if V_i is linked to V_j

Output a kissing-disk drawing of graph Γ

- **Do**
  - **For Each** vertex $u \in V$
    - *Resulting_Forces*=$\sum$ ***Attraction_Forces***$(u) + \sum$ ***Repulsion_Forces***(u)
    - u=u moved by the Resulting_Forces
  - **Next**
  - Recompute Continuance_Condition:
    $\forall(i,j) \in E, x_{ij} \neq (R_i + R_j) \mp ErrorTolerance$
  - Iteration_Count=Iteration_Count+1
- **Until** (Continuance_Condition=False Or Iteration_Count>MaximumIterations)

- **Attraction_Forces**= $AF_{ij} = k_a \Delta x_{ij}$, if $(I, j) \in E$,
  - k_a = attraction strength factor,
  - $\Delta x_{ij} = Distance\ V_i\ to\ V_j\text{-}RestLength(i,j)$
  - $RestLength(i,j) = R_i + R_j$
- **Repulsion_Forces**= $RF_{ij} = \frac{k_r}{xii}$, for all (I, j) *if* $x_{ij} < RestLength(i,j)$
  - k_r = repulsion strength factor
  - $x_{ij} = Distance\ V_i\ to\ V_j$
  - $RestLength(i,j) = R_i + R_j$

FIGURE 21 abstract bubble diagrams drawn automatically by the force-directed graph-drawing algorithm. Changing the area values changes the diagrams.

§ 3.8.3 Drawing Justified Graphs

Drawing justified graphs [after (Hillier, B., Hanson, J., 1984), using Algorithm 3 allows designers to see a configuration literally from different point of views. A justified graph drawing shows the concept of graph theoretical distance (alias depth in Space Syntax jargon). It might help designers more carefully construct spectra such as privacy to community or climatically controlled to uncontrolled spaces. We shall see an educational example of such a usage in the next chapter.

Algorithm 3: a simplified description of Justified-Graph-Drawing; the actual algorithm contains many more details that have been omitted for the sake of clarity and brevity[a]. Minimization of crossings is a WIP.

Given the graph G = (V, E)
Output a depth-justified tree-like drawing of the graph *G*
1- Compute **Breadth-First-Search (BFS)** for the graph Γ, starting from all $u \in V$ called root 2- Store the depth computed by BFS (graph-theoretic distance) of all vertices from root in a list(of lists) 3- For a chosen root, find out how many depth levels are there and which vertices are in which depth levels 4- Draw depth levels as lines and put vertices of each depth on the corresponding depth level 5- For each vertex make a disk of the size attributed to the pertinent space 6- Draw the graph edges 7- Try to minimize crossings

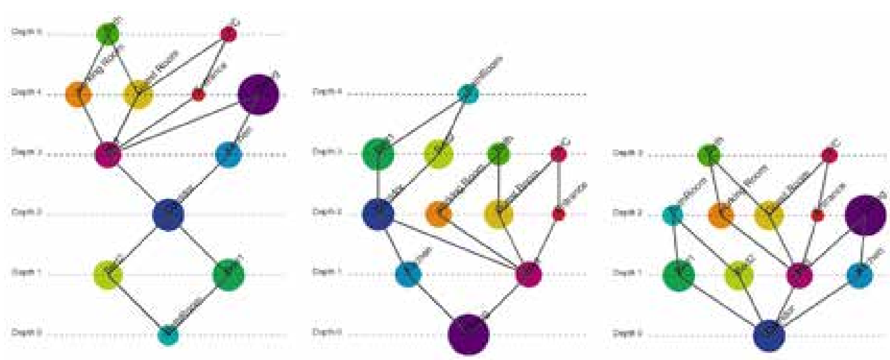


FIGURE 22 justified graphs automatically drawn from an arbitrA0072xy root chosen by designer

a after (Scaffranek, 2012) and (Jiang, B, Claramount C, Klarquist, B, 2000)

§ 3.8.4 Real-Time Space Syntax Analysis

We have chosen Space Syntax as the umbrella structure for the configurational analysis methods used in our design methodology. This is of course because of the architectural relevance of the underpinning principles of Space Syntax theories; particularly those pertained to human perception of space and its links to environmental psychology.

This, however, does not mean that we do not have a critical view on the use of Space Syntax in design analysis. We are aware of some criticisms on a few issues regarding some of Space Syntax indicators such as integration (e.g. see the issue with the so-called diamond value in the formulation of the "Integration" in (Park, 2005) and the problems identified with the so-called "axial lines" in (Ratti, 2004)). The issue that we can identify based on our observation from the usage of the indicators in design practice is the difficulty of interpreting them in plain language and clarifying their exact meaning and their usage in studying the spatial performance of a building. There are also confusions regarding the misuse of centrality indicators as performance indicators. Nevertheless, providing a full account of such details and going to the mathematical depth of these topics diverts our focus and goes beyond the scope of this thesis. For the purpose of this work, we consider Space Syntax indicators as an exemplary analytic engine for architectural configurations. In chapter 6, we offer some alternative methods for spatial network analysis applied in urban configuration analysis.

§ 3.9 Spatial Way-Finding and Geodesics in Buildings

For a smooth introduction to Space Syntax graph theoretical centrality indicators, in the following sections we look at them from the perspective of geodesics or optimal paths. As mentioned before, an architectural configuration can be very abstract at the functional level. Technically, when thinking of a configuration at an initial stage in design it is not even as concrete as an embedded graph; i.e. a graph needs not be drawn to exist. In other words, we might not have any idea or constraint as to how a particular graph can be drawn in 2D or 3D Euclidean space. Therefore, we will deal with configuration graphs at three levels, namely, Graph Theoretical, Topological, and Geometrical.

Dealing with a graph as a set of relations, the notion of shortest path can be reduced to a path consisted of minimum number of nodes. Such a path can be computed using graph traversal algorithms such as Breadth-First-Search (BFS). The interesting point is that Space Syntax research has confirmed that such a distance comes close to something that we might dub ‘mental distance’, which corresponds to the ‘mental map’ that people would construct out of a spatial configuration. A multitude of research papers following Space Syntax approach reveals a connection between graph theoretical measures on graphs constructed from the abovementioned geodesic, namely those making use of *Integration* as means to study urban configurations. Technically, this indicator is formed based on what conventionally called the ‘Graph Theoretical Distance’. It simply tells how many nodes a node in graph is away from another node. For example, in a social network we can conceive of this distance as the number of people that one might need to make their acquaintance in order to connect to some famous person through their connections. In an indoor spatial network, this distance corresponds to ‘the number of rooms one has to pass to get to a destination room’. In this informal definition, we take the notion of room as for spatial node, which can be specified as convex or star-convex shapes.

The so-called first law of geography (Tobler, 1970) has been stated as “Everything is related to everything else, but near things are more related than distant things”. Consistently, the primary idea behind Space Syntax research was that closeness (as derived from the reciprocal of graph theoretical distances) is key to the way people ‘socially’ interact with space and the ways in which some spaces become more public or communal than others. In other words, primary Space Syntax models give us computational insight into the spectrum of privacy-community in spatial networks. The key idea is to look at ‘how far’ is one space to all other spaces in a spatial configuration. This aggregate number is called ‘total depth’ in Space Syntax jargon. If a space is relatively close to other spaces in a system, then it supposedly develops to be a common destination and so it tends to become a communal space. This kind of centrality measures are used to study the so-called “**to-movement**”[a] potentials of spaces within spatial configurations.

Closeness Centrality and Integration (as in Space Syntax) are closely related and based on the distances measured as to the length of some kind of a geodesic (optimal path). Using the geodesics themselves, however, we can think of another type of centrality measures that are based on the number of times a link or a node happens to be part of a geodesic (metric, topological, or any other type of geodesic). If a certain space is often part of optimal paths, then it is reasonable to think that this space has a high potential in terms of “**through-movement**”[b].

a A common Space Syntax term referring to the tendency of a node to attract human movement to itself

b A common Space Syntax term referring to the tendency of a node to attract human movement through itself

§ 3.10 Analysing Architectural Configurations

Centrality measures have been used, mainly by Space Syntax researchers, to study the human movement and socio-functional potential as 'indirect means' to see why some nodes and spaces are more important than others. Using such indicators as means of studying spatial behaviour might seem problematic in some ways; however, centrality indicators are among the best tools available for studying the social and functional potentials of an abstract design representation such as a bubble diagram. We assume that the structure of space has an effect on the patterns of movement in space; and if so, the way we can approach studying such influences could be through graph theory (network analysis), especially in an early stage of design when a configuration does not have enough geometric details to be analysed otherwise.

A valid criticism on the use of centrality indicators (e.g. integration) for studying spatial performance, however, would be the direct usage of a centrality measure as a 'model' of human movement. It must be noted that centrality models can only reveal structural properties of graphs. We might find associations between the dynamics of phenomena taking place within a graph space with centrality indicators, but centrality indicators should not be mistaken with behavioural models. Take for instance closeness centrality (Sabudussi, 1966) and its variant in Space Syntax (Integration): the reciprocal of the sum of distances to other nodes seems to have influence on some community structures, as those nodes with less such distance are more central than others are. It is clear that such indicators only tell something about the structure of the network. Explaining an association between the network structure and the dynamics of such phenomena requires the insertion of more assumptions. This is where the role of behavioural sciences such as environmental psychology becomes essential in developing such indicators.

In case of betweenness centrality (Freeman, 1977), what it indicates is nothing but how many times a node/link happens to be on a geodesic. If we take this as an indication of how many times people are likely to pass by a certain location then we have implicitly assumed that people pass through the type of geodesics that we have modelled. In other words, the betweenness centrality measure is only indicating a structural property of the graph understudy. However, this measure and its underlying assumptions seem to work in studying some socio-economic distributions.

A plethora of studies has found this measure to be associated with such activities as retail businesses and human movement (subject to scale of analysis); namely (Hillier, B., Penn, A., Hanson, J., Grajewski, T. & Xu, J., 1993), (Penn, A., Hillier, B., Banister, D. and Xu, J. , 1998) & (Hillier, B., & Ida, S., 2007).

An alternative approach in studying spatial configurations and their dynamics (e.g. human movement in space) is to develop probabilistic models, for example using stochastic models such as Markov chains. Note that it is possible to convert a measure like betweenness into a probability distribution, e.g. (Volchenkov, D., and Ph Blanchard., 2007).

Let us see how network analysis methods can be used in an 'architectural design process': once the graph is formed, it can be analysed in terms of its potentials as a social-spatial network, for example by Space Syntax measures[a]. These measures, as implemented in our tool suite, are:

- Integration (Hillier, B., Hanson, J., 1984),
- Control (Hillier, B., Burdett, R., Peponis, J., Penn, A., 1987),
- Entropy (Turner, 2007) (based on information entropy introduced in (Shannon, C.E. and Weaver, W., 1949)),
- Choice (betweenness centrality (Freeman, 1977)) for individual functional spaces, and
- Difference Factor (Hanson, 1998) for the whole configuration.

In addition, **Justified Graphs** (as in Figure 22) will be automatically drawn from a node as root to show the same configuration literally from different point of views. Justified graphs might have very practical uses in analysis of relatively large and complex building designs, especially when some hierarchies or sequences are sought as not only spatial Depth (or graph-theoretical distance) but also spectra of other qualities such as controlled temperature.

a Note that there are also Space Syntax measures pertaining to geometric qualities such as visibility as in Visibility Graph Analysis developed by late Alasdair Turner (Turner, 2007). When dealing only with a graph at the most abstract level, these measures cannot be used.

FIGURE 23 exemplary results of Space Syntax analyses for the configuration graph shown in Figure 19

§ 3.10.1 Depth (Space Syntax variant of Graph Theoretical Distance)

The first thing we need to know about a configuration is the distance of each space from any other one in terms of spatial steps (i.e. the spaces in between the two nodes). A distance measured between two nodes, using a graph traversal algorithm such as Breadth-First-Search BFS, on a graph is called the graph theoretical distance between them. We have developed an automated "Justified Graph" drawing tool that visualizes such distances on depth levels. In any configuration, one can choose a point of view to look at their proposed configuration literally from different points of views (see Figure 22).

§ 3.10.2 Integration (Space Syntax variant of Closeness Centrality)

Integration is a measure of centrality that indicates how likely it is for a space to be private or communal. The more integrated a space, the closer it is to all other nodes in a configuration. Integration is calculated by computing the total depth (distance) of a node when the depths (distances) of all other nodes are projected on it. It is formalized as in equation (5), in which k denotes the number of nodes, TD is the total depth as explained above, and D_k, the so-called diamond value is obtained from equation (6)[31]. It indicates how an individual space is private or communal within a configuration .

$$I = \frac{D_k(k\text{-}2)(k\text{-}1)}{2(TD\text{-}k+1)} \quad (5)$$

$$D_k = \frac{2\left(k\left(log_2\left(\frac{(k+2)}{3}\right)\text{-}1\right)+1\right)}{(k\text{-}1)(k+1)} \quad (6)$$

§ 3.10.3 Difference Factor

Difference Factor (Hanson, 1998) is a measure of 'spatial articulation' for a whole configuration. It indicates how differentiated the spaces are within a configuration. It is calculated according to the following equations based on the notion of Relative Asymmetry [in a spatial configuration graph] (denoted as RA) [from (Hillier, B., Hanson, J., 1984)]:

$$RA = \frac{2(TD\text{-}k+1)}{(k\text{-}2)(k\text{-}1)} \quad (7)$$

$$a = the\ maximumm\ RA, b = mean\ RA, c = minimum\ RA \quad (8)$$

$$t = a + b + c$$

$$H: Unrelativized\ Difference\ factor$$

$$H = \text{-}\left\{\left[\frac{a}{t}\ln\left(\frac{a}{t}\right)\right] + \left[\frac{b}{t}\ln\left(\frac{b}{t}\right)\right] + \left[\frac{c}{t}\ln\left(\frac{c}{t}\right)\right]\right\} \quad (9)$$

$$H^*: Relativized\ Difference\ factor$$

$$H^* = \frac{H\text{-}\ln 2}{\ln 3 - \ln 2} \quad (10)$$

§ 3.10.4 Entropy (Space Syntax variant of Shannon Index of Information Entropy)

Entropy values, as described in (Hillier, B., Hanson, J., 1984) and specified in (Turner, 2007), intuitively describe the difficulty of getting to other spaces from a certain space. In other words, the higher the entropy value, the more difficult it is to reach other spaces from that space and vice-versa. We compute the spatial entropy of the i^{th} node as S_i using the i^{th} point depth set:

$$S_i = - \sum_{d=1}^{d_{max}} P_d \, log_2 \, P_d \tag{11}$$

"The term d_{max} is the maximum depth from vertex v_i and P_d is the frequency of point depth "d" from the vertex" (ibid). Technically, we compute it using the function below, which itself uses some outputs and by-products from previous calculations:

Algorithm 4: Entropy Computation

Given the graph $\Gamma = (V, E)$ *as list of list of* $E = (V_i, V_j)$ if V_i is linked to V_j (adjacency lists), *Depths* as List (of (List of integer)), *DepthMap* as Dictionary of integers

Output a list of Entropy indices corresponding to the nodes

- *Initialize Entropies* as List(double)
- **For** *node* as integer in range [0, |V|)
  - integer *How_Many_of_D*=0
  - double *S_node*=0
  - **For** *depth* as integer in range [1, *Depths*[*node*].Max()]
    - *How_Many_of_D*=*DepthMap*.Branch[(*node*,*depth*)].Count
    - double *frequency*= *How_Many_of_D*/|V|
    - *S_node = S_node - frequency * Math.Log(frequency, 2)*
  - **Next**
  - *Entropies* [node] = S_node
- **Next**
- **Return** Entropies;

SourceCode 1: exemplary implementation of Algorithm 4 in C#.NET

```
public List<double> PointDepthEntropy(List<List<int>> G, List<List<int>>
Depths, DataTree<int> DepthMap)
        {
            if (G == null || Depths == null || Depths.Count != G.Count ||
DepthMap == null)
                return null;
            int nodecount = G.Count;
            double[] Entropy = new double[nodecount];
            for (int node = 0; node <= nodecount; node++)
            {
                int howmany_of_D = 0;
                double S_node = 0;
                for (int depth = 1; depth <= Depths[node].ToArray().Max();
depth++)
                {
                    howmany_of_D = DepthMap.Branch(new GH_Path(node,
depth)).Count;
                    double frequency = howmany_of_D / nodecount;
                    S_node = S_node - frequency * Math.Log(frequency, 2);
                }
                Entropy[node] = S_node;
            }
            return Entropy.ToList();
        }
```

§ 3.10.5 Control

Control value (Hillier, B., Hanson, J., 1984), (Hillier, B., Burdett, R., Peponis, J., Penn, A., 1987), specified in (Turner, 2007) (as in equation (12)) intuitively indicates how strongly a vertex in a graph (a space in a configuration) is linked to other points in a superior manner. It is “computed by summing the reciprocals of the neighbourhood sizes adjoining the vertex” (Turner, 2007)) in which D_i is the degree of a ‘neighbour’ node, and n is the number of all neighbor nodes.

$$Control = \sum_{i=1}^{n} \frac{1}{D_i} \quad (12)$$

§ 3.10.6 Choice (Space Syntax variant of Betweenness Centrality)

Choice or Betweenness (Originally introduced as Betweenness by Freeman (Freeman, 1977)) is a measure of importance of a node within a configuration. That literally tells how many times a node happens to be in the shortest paths between all other nodes. It can also be computed for the links connecting the nodes in a similar way. It is computed based on equation (13), in which $\sigma_{jk}(P_i)$ is the number of shortest paths between nodes P_j and P_k which contain node P_i, and σ_{jk} is the number of all geodesics (optimal paths) between P_j and P_k. In social network analysis, the number of shortest paths between two persons is often more than one (easy to imagine why), but this situation almost never happens in spatial network analysis. Therefore, the following equation is a simplified version of that of Freeman (ibid.).

$$C_B(P_i) = \sum_j \sum_k \frac{\sigma_{jk}(P_i)}{\sigma_{jk}} \; (j < k) \quad (13)$$

§ 3.10.7 Random Walk Value (probability of presence)

It can be shown that the stationary probability distribution of a random walk defined on the configuration graph has a probability distribution proportionate to the degree distribution (Volchenkov, D., and Ph Blanchard., 2007). We shall discuss random walks and Markov chains in detail in chapter 5; here we give a hint as to what this measure says.

The intuitive meaning of this measure, when interpreted in terms of the dynamics of a random walk, would be the probability of finding a random walker (a walking drunkard) at a node after a long time of walking. An alternative interpretation would be the 'expected value', or the steady state probability distribution of the spatial system seen as a Markov chain whose dynamics are modelled in terms of the unbiased transition probability of a random walker from a node to a neighbour node (i.e. $Pr_{(i,j)} = 1/\deg(\mathrm{i})$). Note that this transition probability is computed in the absence of any attractions or biases towards nodes. Since the sum of all node degrees in a graph is equal to the number of links times two; the distribution below can be thought of as the node degree distribution relativized as to the sum of degrees.

$$\pi_i = \frac{deg(i)}{2|E|} \quad (14)$$

§ 3.10.8 Analysis versus Evaluation of Spatial Performance

Space Syntax measures and their distributions are qualitatively interpretable into concepts such as privacy and community (Hillier, 2007, p. 22). In case of residential plans for instance, the various representations and measures of Space Syntax show how domestic space manifests life styles, social meanings, and identities of different sub-groups within society (Hanson, 1998). Using Space Syntax methodology, the system interprets spatial arrangement from the very moment it is put in and drawn as a bubble diagram, and gives qualitative feedback on spatial properties and social/functional potentials of the whole configuration in terms of accessibilities, centralities, and likelihood of passage through spaces. According to the design context, designers can interpret these spatial measures into the 'likely' social/programmatic performance of their configurational ideas. As a result, performance analysis is automated by the system; but performance evaluation, i.e. judging the relative goodness of design alternatives, due to the intellectual sophistication of the matter and especially because of its contextual essence, is intentionally left for human designers using the system.

§ 3.11 Synthesising Architectural Configurations

The synthetic process in our methodology commences with embedding the configuration graph put in by the user. It can be thought of, metaphorically, as the process of putting a configuration graph into a geometric shape (see Figure 24 & Figure 25). Note that at phase 1 in Figure 24, there is only a single alternative at hand, and then at the end of phase 2, we will have several alternatives, among which we might want to choose the one that has the best aptitude in terms of an optimization criterion[32]. Later in phase 3, we can potentially produce m different dimensionless rectangular cell configurations for every single such topological layout, each of which permits several dimensioned plan layout patterns. Therefore, we could imagine that the total number of alternatives would be in the order of $n \times m \times o$ solutions. We have only developed the algorithms of phase 0, 1, and 2; but shown how the whole process would be, i.e. if it were to go in the direction defined by March, Steadman, Roth, Hashimshony, and Wachman ((March, L, Steadman, P, 1974), (Steadman, 1983), (Roth, J, Hashimshony,R, Wachman, A, 1982).

As an alternative to rectangular plan layouts (phases 3 & 4 in Figure 24), we have devised free form Computational Geometry constructs (*Isovist Bubbles*) to allow for generic layout processes.

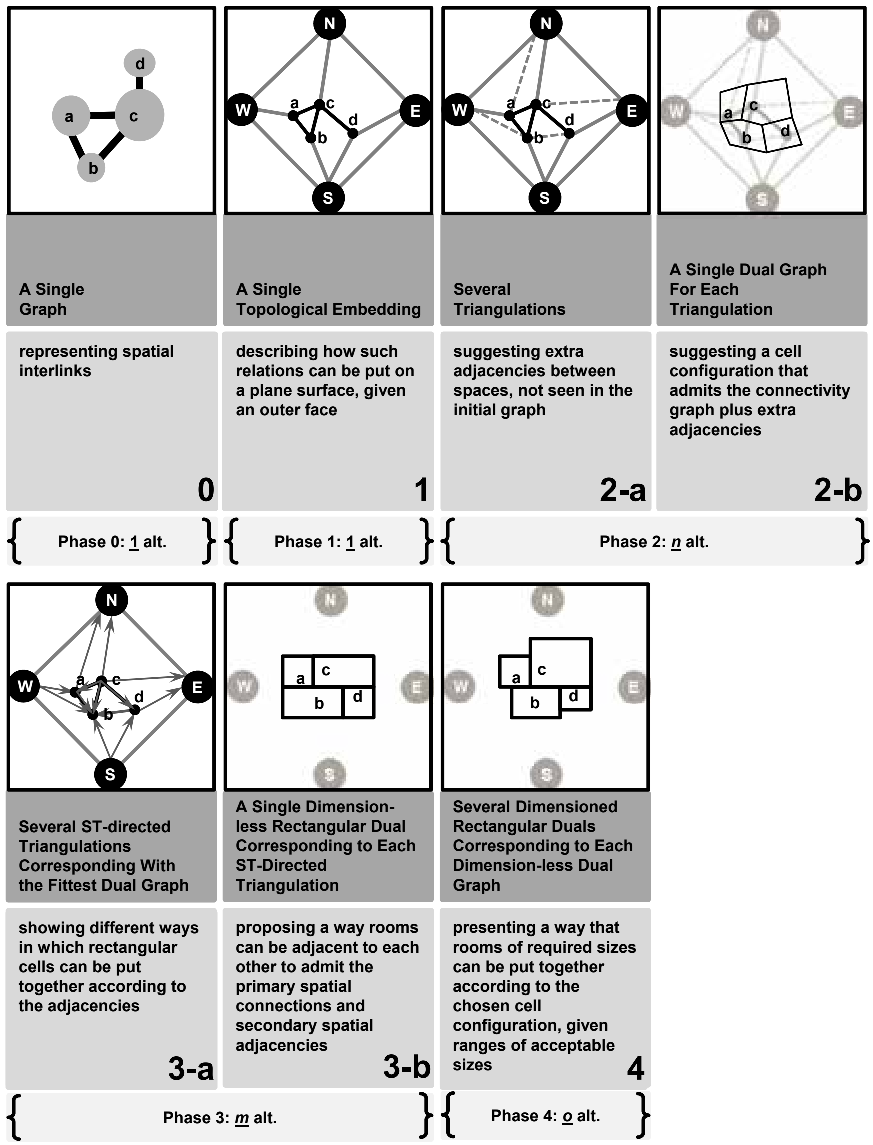


FIGURE 24 a schematic phase model of the proposed design methodology, only the first 2 phases have been implemented algorithmically. Stages 3-a, 3-b and 4 are not included in our proposed methodology.

Although based on a particular geometric constraint, the number of rectangular plan layout typologies can be seen to grow rather rapidly in this process. This is theoretically interesting as it demonstrates the complexity of a spatial layout process. Even if we stick to a single connectivity pattern, the number of geometric possibilities admitting that pattern would be many (perhaps not too many, if adopting such a geometric constraint). For reasons explained later (§ 3.11.6), we abandoned the idea of rectangular plan layout and focused on two Computational Geometry constructs (isovist bubbles), which can potentially produce flexible solutions out of topological plan-layout patterns (see Figure 25).

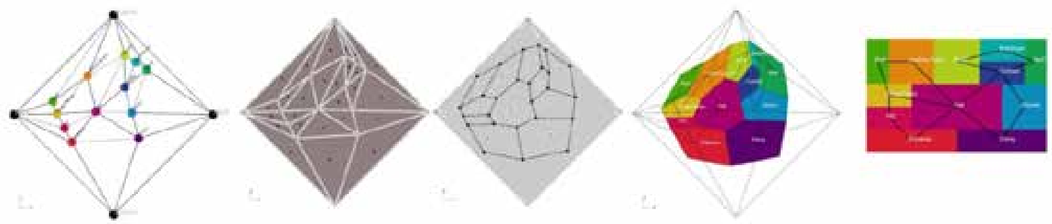


FIGURE 25 The course of computational procedures for triangulating a connectivity graph by adding adjacency links, finding a dual graph and a rectangular dimension-less plan layout pattern. The last step (rectangular drawing) is done with unstable/non-standard algorithms, only for giving an idea as to where the process could go.

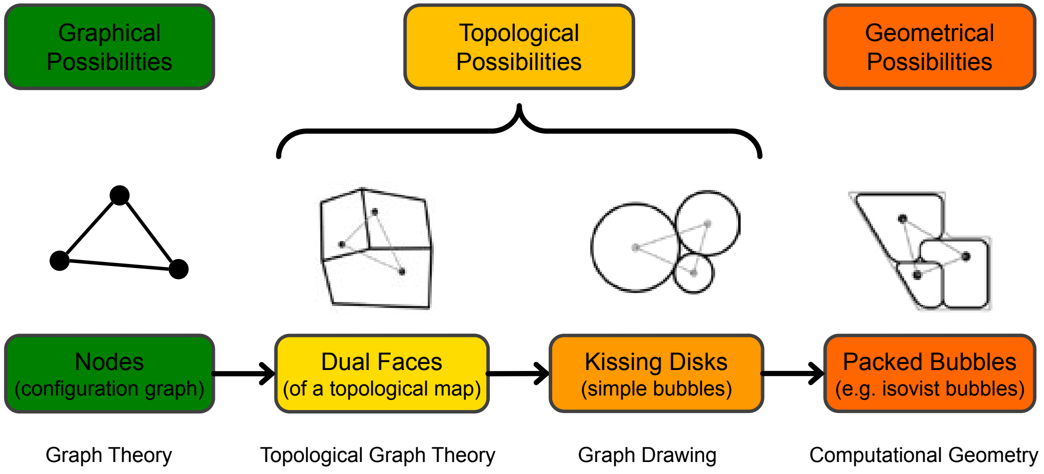


FIGURE 26 the elements of design representations gradually evolve from abstract nodes, without any geometric connotation, to isovist bubbles that have the property that match with the area requirements, the plan boundary, plus complying with the desired spatial connectivity.

We need to deepen our explanation of topological embedding to make our focal point clearer to the general reader. A graph, as mentioned before, does not need to have a geometric representation to exist. For instance, nowadays, with the experience of social networks everybody can realize that the social network does not need to be represented as a geometric shape to exist, however it can be drawn as such, if desired. In other words, a drawing of a graph is not the same as the graph itself. Topological graph theory, studies the embedding (loosely speaking, drawing) of graphs on surfaces (and other topological spaces). According to the preliminaries given before (§ 3.5) a graph Γ does not contain any topologic information per se, whereas a map M does, i.e. the set of its faces F. Technically, a map is a topological representation of a planar graph, i.e. a graph that can be drawn on a plane (or a 2-manifold) without crossing edges (Baglivo, J.A. & Graver, J. E., 1983).

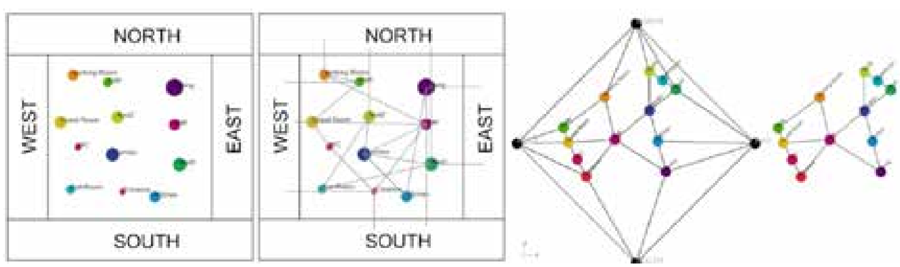


FIGURE 27 from left to right, the sketchpad, a sample configuration graph drawn by a designer (as point nodes and line links) the unique untangled planar drawing of the configuration including and excluding the nominal North-East-South-West sides of the configuration.

The Whitney theorem states: "a planar graph, which is 3-connected has only one choice for the set of faces, i.e. it yields only one map [say one topology, loosely speaking, before further triangulations[a]]" (Baglivo, J.A. & Graver, J. E., 1983). A 3-connected graph is a graph, which cannot be made disjoint unless at least three vertices are removed from it together with their incident edges.

We can find this embedding (for a 3-connected graph) using the Tutte algorithm ((Tutte, 1963)) which is guaranteed to find this unique embedding as a crossing-free straight-line drawing, given a convex face considered as the outer face and fixed. The Tutte Drawing algorithm puts each vertex at the barycentre of its neighbours, i.e. the average position of its neighbours. This drawing has the property that all its faces are convex, including the outer face. This solution corresponds to a unique solution to a system of linear equations (Tutte, 1963). This theorem and the method for creating a Tutte drawing is believed to have made a revolution in the course of development

a To imagine this, consider the embedding on a sphere instead of a plane, this way, no matter which face is considered as the 'fixed' outside, the total set of faces will be the same.

of [topological] graph theory as it strongly states that if the input graph is planar and 3-connected, then the drawing output by the barycentre algorithm is planar, and every face is convex (Eades, 2012).We have made use of this algorithm in a novel way in our methodology, in order to have a solid basis for further investigations on the topological possibilities. By offering a graph formation algorithm, which produces a sketchpad (see Figure 28), once the designer inputs nodes for a connectivity graph, we automatically add four vertices named North, East, West and South (NEWS) to represent the nominal sides of a generic plan, this is inspired by the work of (March, L, Steadman, P, 1974) and (Steadman, 1983)). The idea is that, whatever the plan, it has some kind of connection with the outside world and that it has sides, these vertices generically represent nominal geographical sides, which could be there in lieu of whatever other features outside the plan[33]. The user should provide some additional connections to these sides in addition to the internal connectivity links. These vertices are automatically considered as the fixed vertices of the embedding so the user does not need to introduce any other constraint. After the embedding is done, the algorithm separates the internal connectivity sub graph from the complete graph including the NEWS vertices (Figure 29). Note that the condition of having a 3-connected graph applies to the complete graph containing NEWS vertices, meaning that it does not add any additional constraint to the design of internal configuration. In other words, the internal sub graph can be even 1-connected (see Figure 28 & Figure 29). The designer is automatically prompted to provide enough links with the NEWS vertices such that the whole graph is 3-connected.

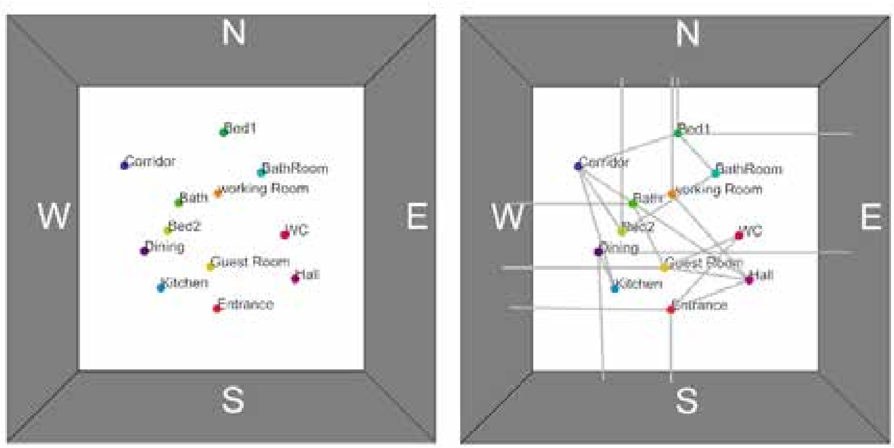


FIGURE 28 The lines drawn by the designer are interpreted into spatial connectivity links and adjacencies with the outer space in nominal NEWS directions

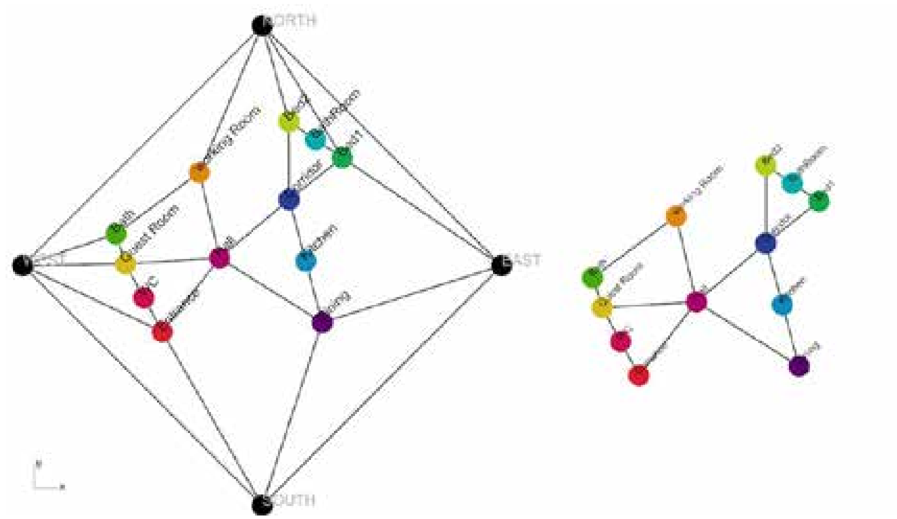


FIGURE 29 a Tutte embedding of the spatial connectivity graph, given NEWS vertices as fixed. This is a 3-connected graph, which has only this map; meaning that this topology is unique.

§ 3.11.1 Producing a Convex Embedding of the Connectivity Graph

A very important method in our methodology is for untangling the embedding of an architectural configuration graph. This method produces a unique topological embedding of the graph on a plane. It is implementing the Tutte algorithm for convex drawing (Tutte, 1963). The valuable point is that once this (linear-time) algorithm converges to an embedding (practically in a tiny fraction of a second for small graphs) we are certain that it is unique. Therefore, that means that no matter how we provide the connectivity input, we always get the one embedding that corresponds to that single graph of connectivity.

A topological embedding indicates how the vertices of a graph are connected to one another on a surface. It is usually expressed in terms of 'face' descriptions. The convex drawing algorithm reveals the unique planar topology of the connectivity graph, given that it is linked in a particular way to the nominal "North, East, West, and South" (NEWS).

A topological description is in between an abstract connectivity description and a concrete geometry. This is exactly a breakthrough in our computational methodology that it uses a Tutte embedding for generating geometric graph drawings and plan layout patterns. This method also indicates if a floor plan is admissible for the set of connectivity requirements; provides an ordering for automated justified graph drawing; and distinguishes a sub graph of the whole connectivity graph (excluding NEWS vertices).

TOC

This sub graph, its vertices and its attributes will be used further on (See Figure 27). Prior to using this method, we check if the connectivity graph can possibly have a planar embedding, i.e. using Euler Characteristic (equation (4)) and one of its corollaries (equation (15)). The Tutte algorithm, however, could deliver result with poor geometric resolution in some cases. To overcome this drawback we introduced our force-directed drawing method in addition. We have also exploited a 'Spectral Graph Drawing' algorithm in chapter 5 (section 5.13) that is closely related to the Tutte algorithm.

Algorithm 5 Tutte's barycentric embedding (explanation based on (Eades, 2012))

Given the graph $G = (V, E)$
Output: a straight-line drawing (embedding) P
Step 1: Choose a subset A of V (to be fixed, we have introduced the virtual vertices of 4 NEWS nominal sides as A) **Step 2:** Choose a location $p_{(a)} = (x_a, y_a)$ for each vertex $a \in A$ (on a convex polygon) **Step 3:** For each vertex $u \in V - A$, place u at the barycentre (average position) of its graph-theoretic neighbours. This can be done using a linear algebraic method for solving a system of linear equations: Let $\boldsymbol{L}$ be the matrix (this is the so-called **Laplacian Matrix** of the Graph G, corresponding to the remaining vertices $V - A$), indexed by $V - A$, and defined by: $$L_{(u,v)} = \begin{cases} deg_{(u)} \; if \; u = v \\ \text{-}1 \; if \; (u,v) \in E \; (if \; u \; is \; adjacent \; to \; v, \; i.e. \; \exists (u,v) \in E \\ \; 0 \; otherwise \end{cases}$$ **Step 4:** Let c and d be the vectors, indexed by $V - A$, defined as: $$c_u = \sum_{w \in A} x_w$$ $$d_u = \sum_{w \in A} y_w$$ $$n = \lvert V\text{-}A \rvert : \boldsymbol{c} = \begin{bmatrix} c_0 \\ \vdots \\ c_u \\ \vdots \\ c_n \end{bmatrix}, \boldsymbol{d} = \begin{bmatrix} d_0 \\ \vdots \\ d_u \\ \vdots \\ d_n \end{bmatrix}$$ **Step 5:** Choose x and y to be the vectors, indexed over the set $V - A$, defined by: $$\boldsymbol{x} = \boldsymbol{L}^{-1}\boldsymbol{c}$$ $$\boldsymbol{y} = \boldsymbol{L}^{-1}\boldsymbol{d}$$ **Step 6:** Choose $p(u) = (x_y, y_u)$ for all $u \in V - A$ and draw edges $(i, j) \in E$

§ 3.11.2 Maximal Planar Graphs and Triangulation

After the unique topology (map) of the connectivity graph is revealed, we can think of the possibilities it implies for different plans admitting such connections. There is one important fact to be observed here: that for two spaces to be connected they need to be adjacent in our definition; however, being adjacent does not necessarily mean being connected. For example, the reader might imagine a kitchen and a bathroom that are adjacent to one another for technical reasons such as passing the water pipes and alike through a shared 'wet' wall. In this case, it is not likely that these two spaces are connected as well. In fact, such adjacencies usually occur because of technical considerations or simply out of compaction. In other words, a connectivity graph implies which spaces 'should be' adjacent to each other, but it does not say which spaces 'could be' adjacent to each other. In order to find such possibilities, the same way we have interpreted connectivity links as edges of the graph, we can add a new set of edges to the graph for the additional adjacencies. The point is that the new graph will necessarily contain the connectivity graph as a sub graph. We can add these new edges up to a level where adding one extra edge will render the graph non-planar. For knowing how many additional edges can be added, we can refer to the Euler characteristic (equation (4)) for an upper bound. Given the fact that in a graph defined in (equation (2)) there can be a maximum of edges above which the graph will be necessarily non-planar (equation (15)).

$$|E| \leq 3|V| - 6 \qquad (15)$$

A planar graph with the number of edges $|E| = 3|V| - 6$ is called a maximal planar graph, which is necessarily a triangulation, meaning that all its faces are triangles. Now, the question is in how many ways these new edges can be added. It is clear that the number of possibilities for triangulating this map has to do with the number of topologic possibilities for the plans; but the question is how are these two entities related?

§ 3.11.3 Finding Dual Spatial Layout Topologies

A convex Tutte drawing can be 'triangulated' so as to give rise to dual graphs that can represent a cell configuration admitting the connectivity graph in itself. While triangulating, we may add links that imply adjacencies that may arise out of compactness and/or enclosure geometric constraints. If we confine the triangulations to a particular type of triangulations, then we may get rectangular dual graphs that can be viewed as dimension-less plan-layout patterns (See Figure 25). Such “dimension-less dissections can be later dimensioned by means of two algorithms introduced in (Steadman, 1983), (Roth, J, Hashimshony,R, Wachman, A, 1982), (Roth, J & Hashimshony, R, 1988), and (March, L, Steadman, P, 1974)[34].

Instead of a technical description, here we give an intuitive definition of dual graphs. A reader familiar with the theory of Space Syntax knows that a plan can be analysed as a graph containing its convex cells and the connectivity relations among them (we explained this point in terms of Poincare duality theorem in the preliminaries of this chapter). This graph, which is constructed out of a cell division, is a sub graph of the dual representation of that cell division. Duality is mutual of course, meaning that given a connectivity graph, a cell division that admits that graph because of cell adjacencies is a dual of a graph that contains the connectivity graph as a sub graph. If we consider the cell division as a map $M = (V, E, F)$ then each face of this map represents a vertex in its dual graph, which contains the adjacency links as well as connectivity links (see Figure 30 & Figure 31) (after (Steadman, 1983)). This implies that if we manage to find all possible triangulations of a connectivity graph, we can reveal all possible topological maps of plans that can admit such connectivity patterns as their dual cell divisions (see Figure 30). Note that these distinct possibilities are plan topologies and should not be mistaken by plan geometries.

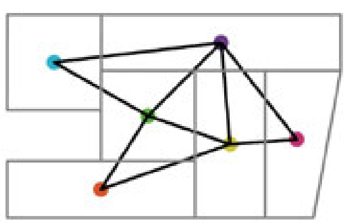


FIGURE 30 a planar cell division and its dual graph (which might contain both adjacency and connectivity links)

§ 3.11.4 Finding All Possible Adjacencies as Maximal Planar Graphs

In computational design practice, there is a trend to randomly generate many geometric possibilities and then use search algorithms to find the best ones according to some formulated goals. Instead, in this work we propose that the computational process could be a process of revealing distinct possibilities in a systematic manner, i.e. revealing the “Design Space” as a **Catalogue of Possibilities**. We prefer to end the search in the domain of topological embeddings of connectivity graphs as we deem each topological embedding "a distinct possibility" that can be further concretized as a plan. We do not follow any optimization path, as we do not have any intention to automate the design process, whatsoever. We can argue that all of the revealed possibilities originate from the configurational idea of the designer, as they are all based on the initial architectural configuration graph. This is instead a theoretical investigation into the matter of syntactic design, i.e. the number of topologic possibilities in a layout process. Interestingly, the possibilities are not 'too many'. In order to generate a catalogue of the mentioned possibilities, we need to perform two computations sequentially:

- Generate all possible triangulations (maximal planar graphs)
- Find the dual cell-configuration of each triangulation

Now, let us go back to the question we posed on the number of ways for drawing a cell configuration corresponding to a configuration graph (§ 3.8 Configurative Design Process). This question now can be reinterpreted as follows: How many triangulations (unique floor plan topologies) are out there? Imagine the connectivity graph embedded by the Tutte algorithm in Figure 29. If we are to triangulate this map, we should look at the non-triangular faces of it. Each non-triangular face can be triangulated in a number of ways and the number of ways one can triangulate a convex polygon with n sides is known to follow the so-called Catalan number of two lower orders C_{n-2} as below:

$$Catalan\ number\colon\ C_n = \frac{1}{(n+1)}\binom{2n}{n} \xrightarrow{yields} C_{n\text{-}2} = \frac{1}{(n\text{-}1)}\binom{2n\text{-}4}{n\text{-}2} \tag{16}$$

The above equation is adopted from (Hurtado, F., Noy, M., 1996), (Hurtado, Ferran, and Marc Noy, 1999), and (Saracevic, M., Stanimirovic, P., Masovic, S. and Bisevac, E., 2013).See Figure 31, Figure 32, Figure 33, Equation 17 and Algorithm 6.

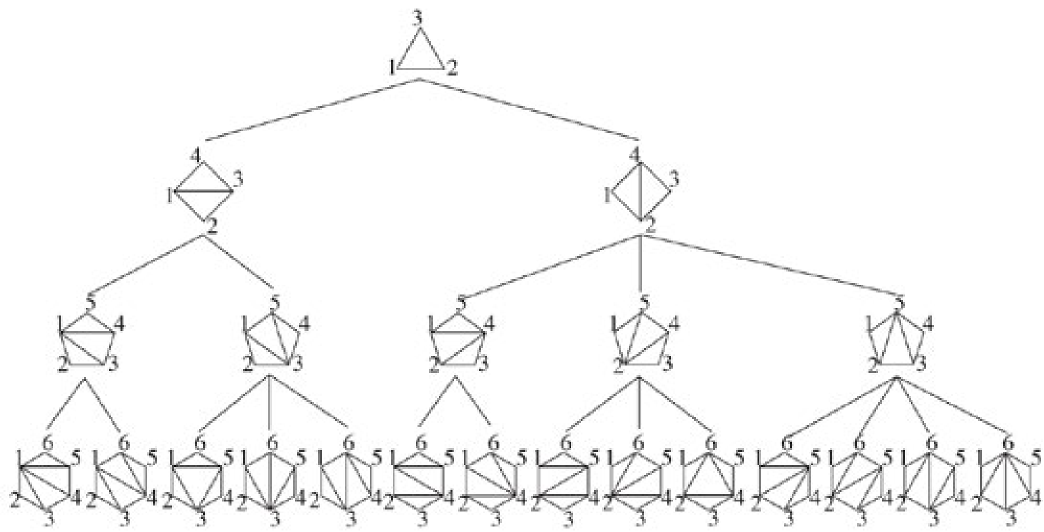


FIGURE 31 from (Hurtado, Ferran, and Marc Noy, 1999) showing the ways in which convex polygons of size smaller than seven can be triangulated.

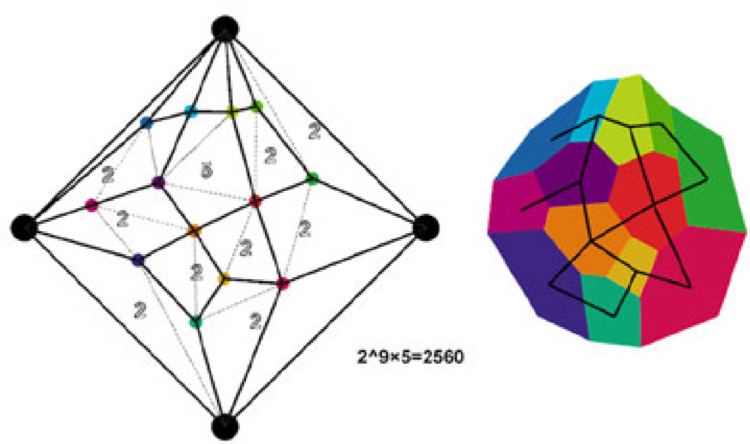


FIGURE 32 triangulation of a convex embedding of a connectivity graph. Impulsive adjacency links are shown as dashed grey lines. A corresponding dual map is shown at the right side, with initial spatial connectivity links shown as thick edges in black. The total number of topologic possibilities pertinent to triangulations is calculated by multiplying the ways all non-triangular polygons can be triangulated, i.e. $\mathbf{2^9 \times 5 = 2560}$. Note that some triangulations might be omitted because a polygon of 5 sides might have been embedded with two adjacent edges in line with one another, this practically converts this polygon into a 4-sided polygon and hence lowers the possibilities.

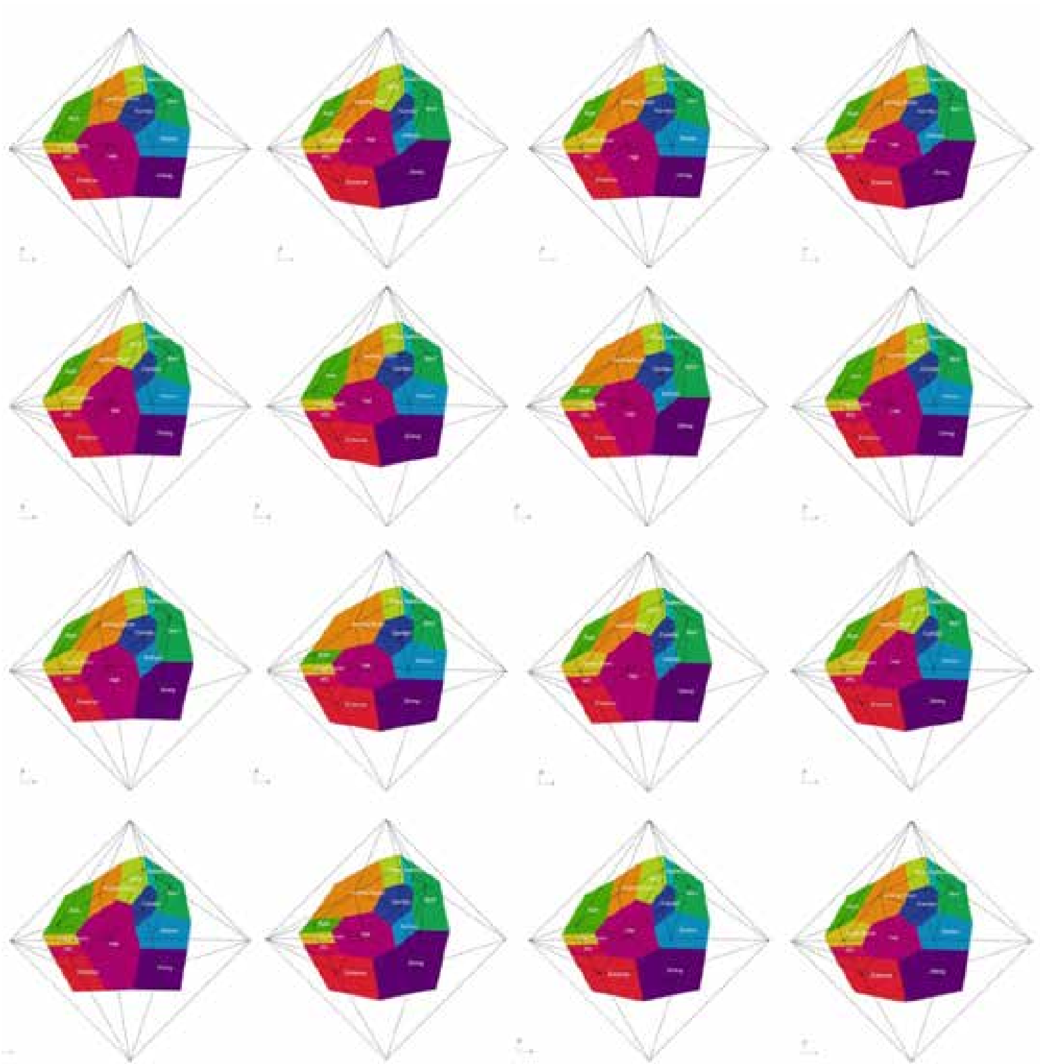


FIGURE 33 shows the 16 feasible plan-layout patterns (duals of 16 possible triangulations of a 2D embedding of a configuration graph) for a sample connectivity graph revealed and enumerated exhaustively.

Through integration with computational design workflows, the user/designer can go further from these topologies and develop various geometries depending on project-specific choices and formal preferences, the possibilities include using our isovist bubbles. See Figure 34, Figure 35, Figure 36, and Figure 37 for an animation-like account of the process.

Finding all triangulations algorithmically is not as easy as drawing them by hand. We have devised the recursive Algorithm 6[a] for enumerating these triangulations inspired by a decomposition of the Catalan numbers, originally conceived of in conjunction with the problem of triangulating polygons by Leonhard Euler, explained in an enlightening lecture by Norman J. Wildberger (Wildberger, 2013). He makes a connection between the Euler's formulation of the problem of triangulating polygons and the Segner's[b] decomposition of the Catalan numbers:

$$Catalan\ number:\ C_n = \sum_{k=0}^{n-1} C_k\, C_{n-k-1} \tag{17}$$

Our algorithm find all triangulations for each non-triangular polygon and then cross-reference them to find all triangulations corresponding to the whole map.

a A detailed description of this sophisticated algorithm would take us far away from the scope of this thesis, and hence, it is postponed to a future publication.

b Johann Andreas Segner (1704-1777), a Hungarian scientist.

Algorithm 6: Find All Possible Triangulations of a Sub-Maximal Planar Graph

Input a polygonal mesh PM// that is a planar embedding of a planar graph

Output a list of triangulated meshes// that contains all possible triangulation maps, which can make the initial embedding a maximal planar graph

- ❖ **Procedure** List(Mesh) ***EnumerateTriangulations***(polygonal mesh PM){
  - ➢ Form set $\mathbb{T}^n$;// that is the Cartesian product of all tessellation sets
  - ➢ $\mathbb{T}^n = T_1 \times T_2 \times \ldots \times T_n \mid T_i := \underline{\textbf{\textit{TriangulatePolygon}}}(PM.\mathcal{F}_i)$;
  - ➢ $//T_i = \{\forall \mathcal{T}(\mathcal{F}_i) | tesselation\ \mathcal{T}: is\ maximal\ planar\ of\ i^{th}\ polygonal\ face\}$
  - ➢ List(Mesh) ALL_TINS=new List(Mesh);
  - ➢ **For each** ($\mathcal{T} \in \mathbb{T}^n$){
    - ▪ ALL_TINS.Add(PM.Vertices, $\mathcal{T}$);// $\mathcal{T}$ ≔ a set if triangle faces}
  - ➢ **Return** ALL_TINS;}
- ❖ **Procedure** List(Triangulation) ***TriangulatePolygon***(PolygonFace P)){
  - ➢ **If** (P.IsTriangle()){**Return** [P].ToList();}
  - ➢ List(Triangulation) Triangulations=new List(Triangulation);
  - ➢ **For** (integer ear=2; ear<Polygon.VertexCount; ear++){
    - • List(PolygonFace) LR= ***DividePolygon***(Polygon,ear);
    - • **If** (LR.Count==1){
      - ♦ List(Triangulation) T_Set= ***TriangulatePolygon***(LR[0]);
      - ♦ **For each** (Triangulation T in T_Set){
        - ➢ T.Add(new Triangle(0,1,ear));
        - ➢ Triangulations.Add(T);**}}**
    - • **Else{**
      - ♦ List(Triangulation) L_T= ***TriangulatePolygon***(LR[0]);
      - ♦ List(Triangulation) R_T= ***TriangulatePolygon***(LR[1]);
      - ♦ List(Triangulation) T_Set =new List(Triangulation);
      - ♦ T_Set=L_T×R_T;//Cartesian Product of the two sets
      - ♦ **For each** (Triangulation T in T_Set){
        - ➢ T.Add(new Triangle(0,1,ear));
        - ➢ Triangulations.Add(T);**}}**
  - ➢ **Return** Triangulations;}
- ❖ **Procedure** Mesh Face[] ***DividePolygon***(*Mesh Face Polygon, Integer ear*){
  - ➢ Mesh Face L_Polygon=Polygon to the LHS of triangle {0,1,ear};
  - ➢ Mesh Face R_Polygon=Polygon to the RHS of triangle {0,1,ear};
  - ➢ **If** (L_Polygon.Count<3) {**Return** [R_Polygon];}
  - ➢ **If** (R_Polygon.Count<3) {**Return** [L_Polygon];}
  - ➢ **Return** [L_Polygon,R_Polygon];}

TOC

§ 3.11.5 Topological Possibility versus Geometrical Possibility

Up to this point, we have not considered any geometric property for the nodes corresponding to the functional spaces. Note that speaking of topologic possibilities, there is no specification whatsoever on the geometry of the surface (2-manifold) on which the map is supposed to be laid out. This surface could be a plane surface as well as the surface of a sphere in this sense. However, we know that we want to check possibilities for floor plans, and these plans are to be realized on planes (that is, if the case is confined to 2D configurations). If, for instance, the area associated with a node surrounded by other nodes does not admit the node to be placed there due to lack of space, the possible topology becomes unrealizable in our case. This is an example that suggests a match of desired area values and the area of the convex map is desirable. However, this is not a generic situation, and no general conclusion can be made from this. Therefore, instead of searching for a 'good' map (an optimum map), we divert our attention to what a map might mean in reality[35].

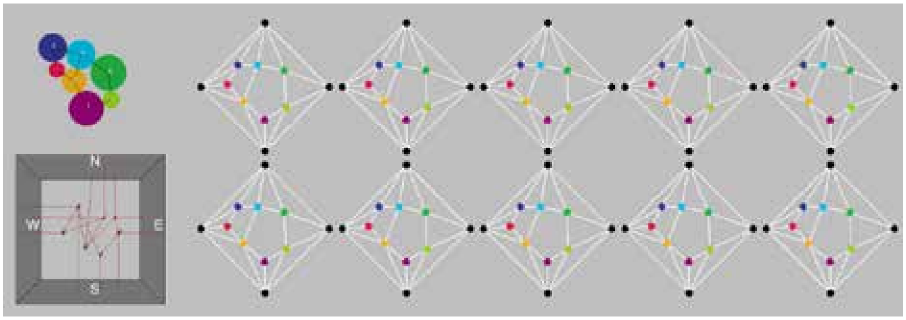


FIGURE 34 shows the bubble diagram representing the spatial interlinks specified in the sketchpad at the left-side of the picture. A single convex map is found using Tutte algorithm by fixing vertices representing NEWS nominal sides of the plan. The convex map has been copied 10 times knowing that there would be 10 possible plan layout topologies coming out of it.

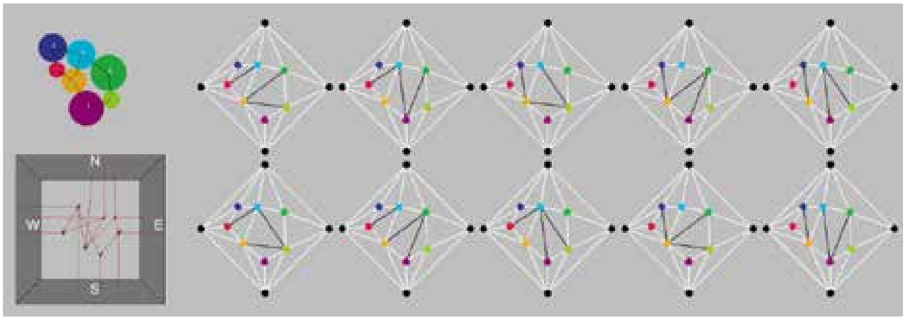


FIGURE 35 shows the 10 possible ways to triangulate the convex map drawn in white lines.

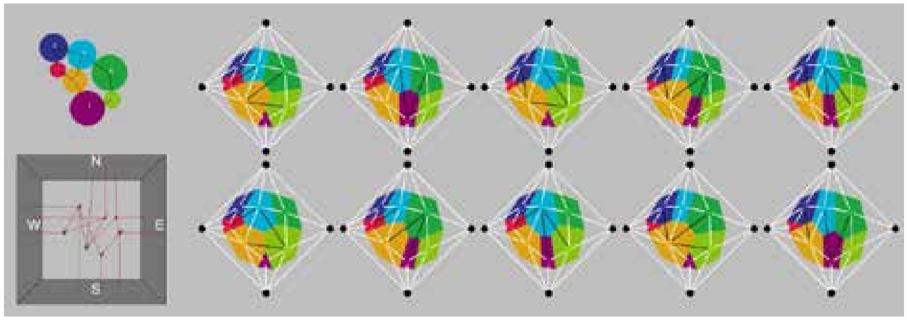


FIGURE 36 shows the dual cell-divisions corresponding to the triangulations found earlier.

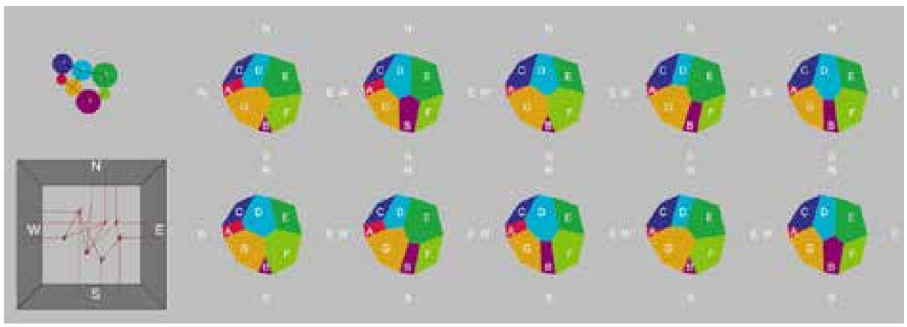


FIGURE 37 only shows the plan layout topologies, without the spatial interlinks. Note that the original connectivity requirements are met in all plan topologies. The difference between these plan topologies is in the adjacencies that have not been specified in the initial bubble diagram.

The plan layout topologies catalogued in the process could be the starting point for devising a concrete and geometric plan layout. Continuing in the direction suggested by Steadman (Steadman, 1983), we could have worked on converting these topologies to "dimensionless rectangular dissections" and dimensioning them. We, however, tried alternative paths, which could possibly result in solutions that are more generic (free form) than rectangular dissections.

§ 3.11.6 A Note on Rectangular Floor Plan Layout

March & Steadman (March, L, Steadman, P, 1974) & (Steadman, 1983) have introduced an elegant way to convert a class of triangulations into a class of drawings called rectangular drawings or rectangular dissections. Here we intend to define a more generic class of possibilities. In comparison, our method (while inheriting a lot from the method of March & Steadman) provides a list of topologic possibilities but does not lead to an automated plan layout generator, instead it provides for an interactive method of exploring geometric possibilities. Theoretically, the geometric possibilities corresponding to one topologic layout are infinite unless a restriction introduced on the type of geometry to be produced. This is of course consistent with the fact that architectural design is a very creative activity given innumerable possibilities for realizing a single idea. In the absence of information regarding the type of shapes desired in a plan, we have focused on packing bubbles as generic ways of interpreting a topologic map into a geometric cell division.

§ 3.11.7 Radical Axis, Power Diagrams, and Alpha Complexes (Shapes)

Having started with the idea of bubble diagrams, led us to have a closer look at the geometry of physical bubbles. In the absence of any geometric assumption, the simplest planar shape with a known area would be a disk/ball (2D). Let us imagine a boundary in which the plan must be drawn and suppose that we have selected a map from the catalogue of possibilities introduced before.

If we think of rooms as bubbles that try to establish themselves within the boundary with a predefined area, they might take a shape that is referred to as an Alpha Shape (a.k.a. alpha complex), closely related to a Weighted Voronoi diagram. A weighted alpha complex is a convex decomposition of a union of such balls (disks), using weighted Voronoi cells. From our perspective, it is also a generic dual of a triangulated map such as a Delaunay triangulation (for an exact definition see (Edelsbrunner, H., & Harer, J., 2008)). The borders between cells in this complex are different from the regular Voronoi diagram in that they are not bisectors but they are radical axes.

A radical axis between two cells (balls of different radii) is a line along which each point has the same 'power' from both cells, where power of a point is a real number h that reflects the relative distance of a given point from a given circle. This is defined as the squared distance of the point from a ball minus the radius of that ball squared (equation (18)).

$$power\ of\ a\ point\ to\ a\ ball\ with\ radius\ r: h = S^2\text{-}R^2 \qquad (18)$$

Given two balls of the same radii, the locus of points whose power is equal to both balls happens to be the bisector of the line between the two centre points of the balls, but if the radii are different, then we will have a situation as shown in Figure 38.

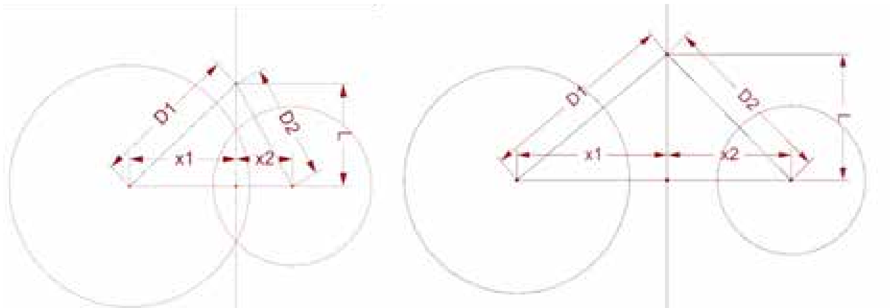


FIGURE 38 shows radical axis between two balls of different sizes. It seems trivial to find the location of the radical axis on the left situation but for the right hand one we need a method.

On a radical axis, the power to both balls should be equal, therefore we have:

$$D_1{}^2 \text{-} r_1{}^2 = D_2{}^2 \text{-} r_2{}^2 \tag{19}$$

$$L^2 + x_1{}^2 \text{-} r_1{}^2 = L^2 + x_2{}^2 \text{-} r_2{}^2 \tag{20}$$

$$x_1{}^2 \text{-} r_1{}^2 = x_2{}^2 \text{-} r_2{}^2 \tag{21}$$

$$x_1 = \frac{1}{2}\left[D + \frac{r_1{}^2 \text{-} r_2{}^2}{D}\right] \;\&\; x_2 = \frac{1}{2}\left[D \text{-} \frac{r_1{}^2 \text{-} r_2{}^2}{D}\right] \tag{22}$$

Algorithm 7: Alpha Complex[36]

Input a boundary, Cells represented by Vertices V with and area values A_i (with Radii $R_i = \sqrt{A_i/\pi}$) and a plane

Output a list of alpha cells (polygons) formed according to powerlines and radii

- ❖ Define Bubbles as a list(of cells)
- ❖ **For Each** vertex $u \in V$
  - ▪ Find the balls that intersect with this ball
  - ▪ **For Each** intersecting ball
    - ♦ Compute the radical axis between them and divide the boundary into two half planes
    - ♦ Choose the half plane corresponding to u
  - ▪ **Next**
  - ▪ VoronoiCell=Boolean Intersection of all half planes corresponding to u
  - ▪ AlphaCell= Boolean Intersection of the ball u with the VoronoiCell
  - ▪ **If** AlphaCell is not dominated by its neighbors **then** Bubbles.Append(AlphaCell)
  - ▪ **End If**
- ❖ **Next**
- ❖ **Return** Bubbles

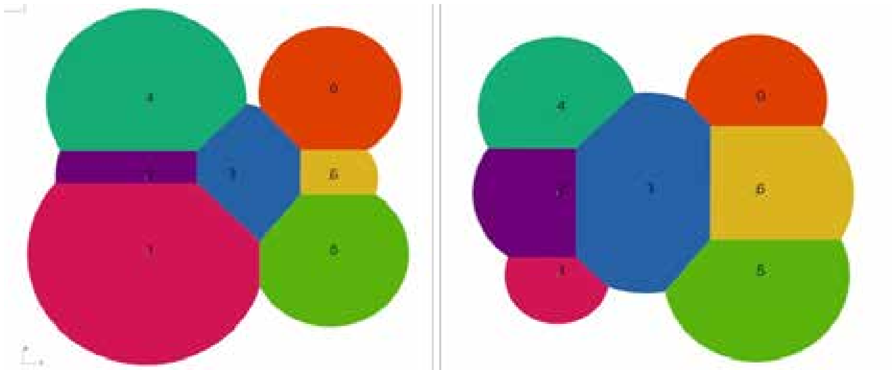


FIGURE 39 generic alpha complexes for one set of vertices with two sets of area values.

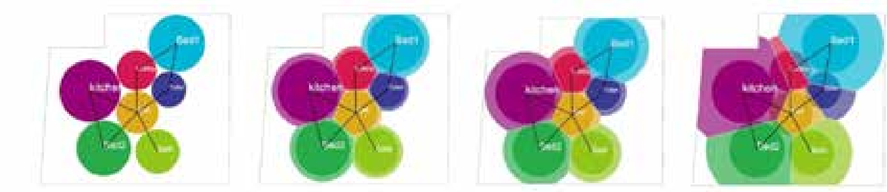


FIGURE 40 a kissing-disk diagram is converted into an alpha complex gradually, simulating the compaction of cells. Note that the cells are always convex before being cut by the boundary; many different shapes even rectangular cells can be made this way but there is no elegant way to make sure the area of a cell after drawing is as much as intended. However, this is a very generic dual representation of a triangulated map.

§ 3.11.8 Towards freeform geometric layouts

For the bubbles of the initial bubble diagram to be better matching with both the local situations in plans (such as boundary and obstacles) and to make sure they will have the intended area, we have devised new spatial units that we call them 'Isovist Bubble 2D' and 'Isovist Bubble 3D'. This was to answer this theoretical question: How do bubbles grow into each other within a confinement, preserving their area and their entirety as visible units of space (ideally preserving their intended connectivity)Isovist bubbles were invented to allow for configurative design of free-from spaces.

Imagine an enclosure that is an **isovist** (Benedikt, 1979), i.e. a 'star-shaped polygon' entirely visible from its centre. Our isovist bubbles have the additional property that they seek to preserve their area (in 2D) or volume (in 3D) wherever they go, i.e. even when dented after colliding with obstacles. An isovist bubble can only be made using computational geometry.

A packing of isovist bubbles would be a class of generic geometric possibilities –i.e. in the absence of further information from the user. The isovist bubbles would change if their centre points move, however they will preserve their area/volume properties and remain as isovist outlines (entirely visible polygons/polyhedrons from their centre points).

Isovist bubbles seem to be more suitable as spatial units for syntactic design compared to convex spaces (as customary in Space Syntax and indoor configuration analysis), as it is very normal to find rooms in plans that are not convex but are star-convex or isovist[a]. Therefore, this definition provides a more generic description of possibilities (Figure 26, Figure 41, and Figure 42).

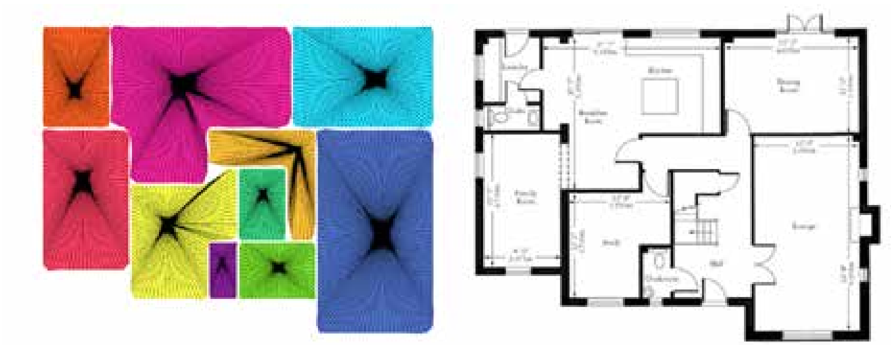


FIGURE 41 isovist bubbles could be seen spatial units of syntactic design corresponding to an example architectural plan. The image is post rationalized (the plan is not generated by Isovist bubbles).

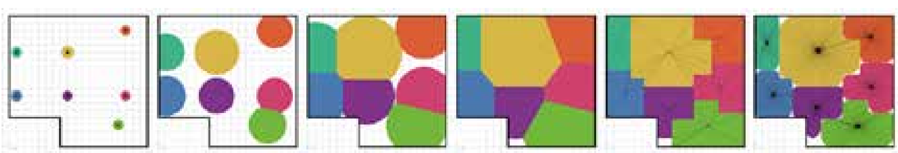


FIGURE 42 an alpha complex converted into an isovist bubble packing. Note that the area values are not exactly the same as the desired values. In order to constraint the shape to the input area values, a different class of alpha complex algorithm needs to be developed. The complex at the right has star-convex cells or isovists, which are entirely visible from their centre points.

The above images (Figure 41, and Figure 42) are produced to help the reader imagine the possibility of using area & visibility preserving agents (Isovist Bubbles) in making spatial layouts. However, the images are post-rationalized. So far, we have only made the smart agents for this process but not the agent-based design process.

a A star convex polygon is a polygon, which has a 'kernel' region from which the entire polygon is visible.

§ 3.12 2D Isovist Bubbles

Suppose a 2D Isovist Bubble as a polygon with n vertices; this polygon is supposed to be of a certain size given as area; but it has another actual area smaller than the desired, because it has hit a set of obstacles. We want to find out a reasonable amount of radial growth for this Isovist Bubble to compensate the area deficiency. For finding this, we use an approximation of the area of the polygon, given that we always have a polygon with vertices nearly distributed circularly such that there is a $\theta = 2\pi/n$ angular span in between every two radial axes. We can then approximate the perimeter by adding up the lengths of arcs corresponding to these radial axes. Assuming an area element for the Isovist polygon as a triangle with the base nearly the same as the aforementioned arc and a height equal to the length of the corresponding radial axis we can approximate the area in terms of radii values. Now, assuming a radial growth for the whole shape, we can think of a R_d as a differential element to be added to current radii. Then the question is how much should this be. We can find this value in terms of A_d the difference in area and the approximated perimeter of the shape by solving a quadratic equation as below:

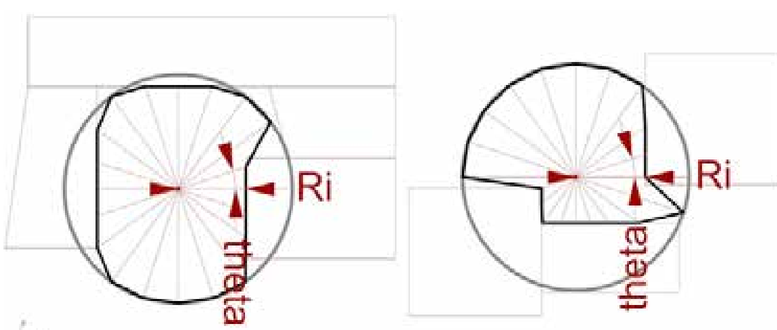


FIGURE 43 an isovist polygon with radial axes in two different situations

The following equations show how during deformation due to collision with obstacles a radius differential is computed to provide for gradual growth of the bubble to preserve its area. Considering r_i as the length or radius of the i^{th} ray shot from the centre to check visibility we can estimate the upper bound for the infinitesimal positive differential to be made in radii for growing the bubble as below:

$$Perimeter\ P \cong \sum_{i=0}^{n-1} r_i\theta = \sum_{i=0}^{n-1} r_i \frac{2\pi}{n} = \frac{2\pi}{n}\sum_{i=0}^{n-1} r_i \quad (23)$$

$$Area\ A \cong \sum_{i=0}^{n-1} \frac{1}{2} p_i r_i = \frac{1}{2}\sum_{i=0}^{n-1} \frac{2\pi r_i}{n} r_i = \frac{\pi}{n}\sum_{i=0}^{n-1} {r_i}^2 \quad (24)$$

$$A + A_d = \frac{\pi}{n}\sum_{i=0}^{n-1} (r_i + r_d)^2 = \frac{\pi}{n}\left[\sum_{i=0}^{n-1} {r_i}^2 + \sum_{i=0}^{n-1} {r_d}^2 + \sum_{i=0}^{n-1} 2r_i r_d\right] \quad (25)$$

$$A + A_d = \frac{\pi}{n}\sum_{i=0}^{n-1} {r_i}^2 + A_d = \frac{\pi}{n}\sum_{i=0}^{n-1} {r_i}^2 + \frac{\pi}{n} n{r_d}^2 + \frac{2\pi}{n} r_d \sum_{i=0}^{n-1} r_i \quad (26)$$

$$A_d = \pi {r_d}^2 + \frac{2\pi}{n}\sum_{i=0}^{n-1} r_i \times r_d \xrightarrow{yields} \pi {r_d}^2 + \frac{2\pi}{n}\sum_{i=0}^{n-1} r_i \times r_d - A_d = 0 \quad (27)$$

$$a = \pi, b = \frac{2\pi}{n}\sum_{i=0}^{n-1} r_i = P, c = -A_d \xrightarrow{yields} r_d = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (28)$$

$$r_d = \frac{-P \pm \sqrt{P^2 + 4\pi A_d}}{2\pi} \xrightarrow{r_d > 0\ and\ \sqrt{P^2 + 8\pi A_d} \geq P} r_d = \frac{-P + \sqrt{P^2 + 4\pi A_d}}{2\pi} \quad (29)$$

Since the area of the shape is always greater than or equal to the estimated value and there are always less than (or equal to) n points for which there can be a radial extension without rendering the shape non-isovist, this r_d is a 'reasonable' approximation of the radial differential extension. In other words, this approximated area is smaller than or equal to the area of both a regular polygon and an isovist polygon.

Algorithm 8: 2D Isovist Bubble (IVB2D)

Input the centre of the 2D Isovist disk, a desired area, a set of 2D obstacles

Output an Isovist (star-shaped) visibility polygon with the desired area

- ❖ **For Each** vertex
  - ▪ Define SightLins as a line(IsovistBubble.Centre,vertex)
  - ▪ **For Each** obstacle
    - ♦ **If** SightLine hits the obstacle **Then**
      - ▪ Store where on the SightLine it hits in *hits*
    - ♦ **End If**
  - ▪ **Next**
  - ▪ Define FirstHit as point=SightLine(at nearest hit)
  - ▪ Replace vertex with FirstHit
- ❖ **Next**
- ❖ IsovistBubble= **New** Polyline(veretexlist)
- ❖ **Do**
  - ▪ **AreaDifference** $\mathbf{A_d}$= initial IsovistBubble.area-IsovistBubble.area
  - ▪ Perimeter p=new Isovist.length
  - ▪ Compute $R_d = \frac{-p+\sqrt{p^2+4\pi A_d}}{2\pi}$
  - ▪ **For Each** vertex in IsovistBubble
  - ▪ **Define** SightLine.Direction=Vector(IsovistBubble.Centre,vertex)
    - ♦ Define MovedVertex as vertex.move(Rd*SightLine.Direction)
    - ♦ **If** MovedVertex is not in any obstacle **Then**
      - ▪ Replace vertex with MovedVertex
    - ♦ **End If**
  - ▪ **Next**
  - ▪ **IsovistBubble=New** Polyline(veretexlist)
- ❖ **Until** $(A_d \geq AreaTolerance)$

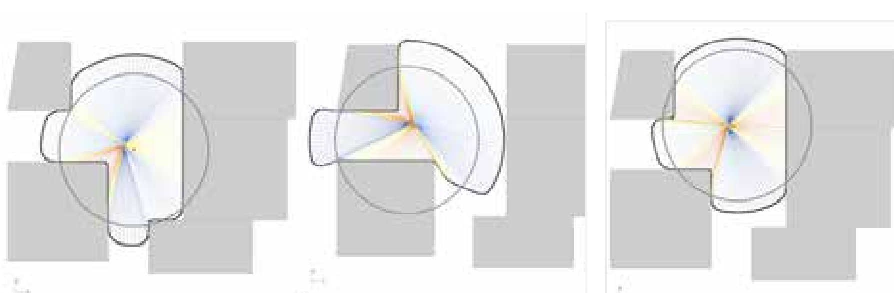


FIGURE 44 Isovist Bubbles made in different situations with the same area as the grey circle, the radial axes have been colour coded as to their distance from the nearest obstacle. A disk as an isovist is transformed into another isovist fit to both free space and the area of the first disk.

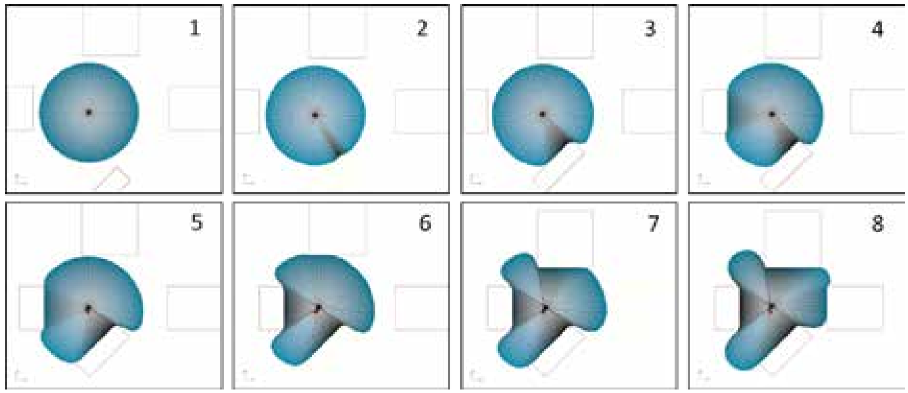


FIGURE 45 shows an Isovist Bubble gradually getting deformed when obstacles getting closer to its centre, shaded black for dents.

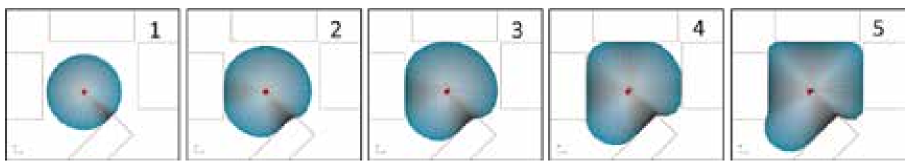


FIGURE 46 shows an Isovist Bubble gradually getting larger while getting dents because of the obstacles around it.

§ 3.13 3D Isovist Bubbles

The same concept of an Isovist bubble, i.e. a bubble that remains an isovist and seeks to reach a desired size can be generalized to 3D. In absence of obstacles, an Isovist Bubble 2D (IVB 2D) would become a disk and an Isovist Bubble 3D (IVB 3D) would become a ball. The concept of a star-convex polygon also generalizes to a star-convex polyhedron. We can only conceive of such entities using computational geometry algorithms.

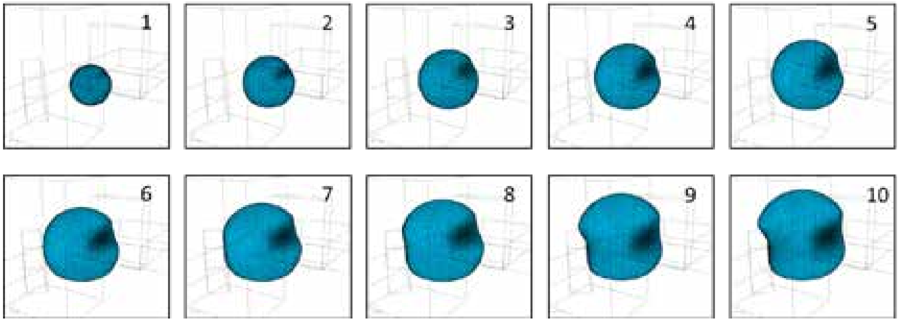


FIGURE 47 shows an Isovist Bubble 3D gradually getting larger in the midst of some obstacles.

Note that the IVBs are shaded black corresponding to the amount of deformation in dents. The hue colour is arbitrary to allow for visualizing rainbow coloured spaces together.

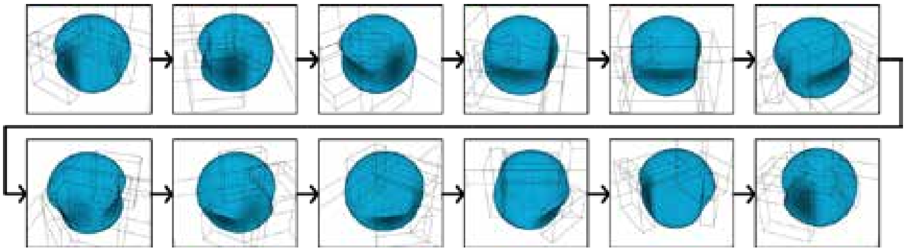


FIGURE 48 goes around an Isovist Bubble 3D to show it from different view angels.

An Isovist Bubble 3D is made up of view rays shut in many directions symmetrically according to a notion of desired volume. Once the rays hit obstacles then a point is assigned as being on the surface of the Isovist Bubble in question. Interpolating these points gives the first shell of the bubble. If the current volume of the shell is not as much as desired then the rays start getting slightly bigger so as to approach the desired volume, while this time the rays are being shot from a new vantage point that is the centre of gravity of the previous iteration of the bubble.

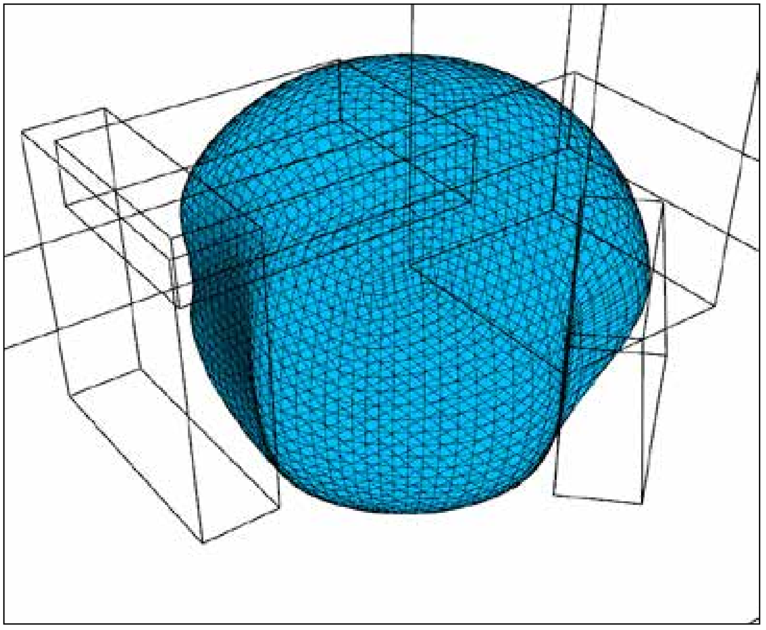


FIGURE 49 shows a close-up of an IVB 3D

Algorithm 9: 3D Isovist Bubble (IVB3D)

Input the centre of the 3D Isovist ball, a desired volume, a set of 3D obstacles

Output an Isovist (star-shaped) visibility polyhedron with the desired volume

- **For Each** vertex
  - Define SightLins as a line(IsovistBubble.Centre,vertex)
  - **For Each** obstacle
    - **If** SightLine hits the obstacle **Then**
      - Store where on the SightLine it hits in *hits*
    - **End If**
  - **Next**
  - Define FirstHit as point=SightLine(at nearest hit)
  - Replace vertex with FirstHit
- **Next**
- IsovistBubble= **New** Mesh(veretexlist)
- **Do**
  - **VolumeDifference** V_d= RequiredVolume-IsovistBubble.Volume
  - A= Isovist.SurfaceArea
  - Compute $dR = \alpha \frac{dV}{A}$, $\alpha \in [0,1]$
  - **For Each** vertex in IsovistBubble
  - **Define** SightLine.Direction=Vector(IsovistBubble.Centre,vertex)
    - Define MovedVertex as vertex.move(dR*SightLine.Direction)
    - **If** MovedVertex is not in any obstacle **Then**
      - Replace vertex with MovedVertex
    - **End If**
  - **Next**
  - **IsovistBubble=New** Mesh(veretexlist)
- **Until** $(dV \geq VolumeTolerance)$

§ 3.14 Discussion: a new way of designing buildings

Our initial idea of a topological layout methodology was based on bubble diagrams conventionally used by designers. We invested on the fact that a conventional bubble diagram is a comprehensible representation for designers for reflecting on spatial configurations, and at the same time, interpretable for computer programs as a configuration graph and that it does not imply a single geometric form. Bubble diagrams convey very important meanings that may not be seen easily by bare eyes; for instance they implicate which spaces are to be relatively private and which ones are to be communal and much more. We find it very important to reveal such meanings from the very beginning of a design process, and report these meanings to the designer so that they can see whether the bubble diagram corresponds to their initial ideas about matters such as privacy and community of spaces.

Our proposed system reads a bubble diagram in an intuitive way and translates it into a configuration graph; it later provides the designer with Space Syntax measures; and eventually explores a particular class of plan layout patterns, which have the same configuration graph represented in the initial bubble diagram. These layout patterns can be used later as starting points by designers to elaborate their plan layouts; potentially concretized into freely shaped geometric forms by using the smart agents called Isovist Bubbles.

The idea of exploring concrete (geometric) plan layout patterns, which share an abstract connectivity pattern (a graph), brings about questions on the nature and the size of the design space, i.e. the catalogue of design possibilities. Theoretically, there could be countless number of plans sharing the same connectivity pattern. However, if we confine the search into a specific class of geometric shapes, such as rectangles, there can be a way to enumerate several alternatives systematically, according to the design inputs.

A workable idea for configurative design, exemplified in our methodology, is to first reach at a topological embedding of a connectivity graph to bring it closer to become a geometric pattern. We have currently achieved this embedding on 2-manifolds (surfaces). An area of future work would be topological embedding of graphs in 3D, using tetrahedrons and tertahedralization algorithms[37]. Throughout this process, in a few steps, the number of possibilities grows rapidly so that we need to select certain paths, methodically, to explore ranges of these possibilities (technically referred to as a design space). Figure 24 depicts a schema of such a design space and the challenge of exploring it systematically.

It is left for the designers to decide on how they want to alter their ideas during the design process, but the methods always provide them with automatic feedback on the configurational properties of what they design; while showing them their own ideas, literally, from different point of views. It is important to note that these ideas usually evolve during the course of design process, as problem formulations and solutions evolve together (Dorst & Cross, 2007). Viewing a justified graph, designer can choose from which space the other spaces are seen, say from different points of views, and analyse it in terms of syntactic measures. This helps designers see if what they have proposed in terms of a bubble diagram actually matches with their initial ideas on privacy/community, spatial articulation and other spatial qualities such as the probability of spatial movements.

We argue that by using our proposed methodology designers can explicitly approach the realization of a spatial configuration that is required for programmatic reasons (such as the examples shown as REL charts for hospitals). Furthermore, they have full intellectual control over the spatial configuration of their designs; they can benefit from computation in seeing their own ideas from different angles; and they receive objective feedback on the spatial qualities and indications on likely programmatic performance of their designs. We have implemented this methodology in a computational design toolkit, which is introduced in the following chapter.

Key innovations in our methodology include the followings:

- Developing a systematic approach for encoding bubble diagrams as labelled configuration graphs and drawing them automatically;
- Using a Tutte diagram to reveal the unique topology of a configuration graph, currently using automatically inserted nominal NEWS sides to a spatial configuration, i.e. as the fixed vertices[38] of a convex outline for it;
- A novel algorithm for enumerating all possible triangulations of a polygonal mesh based on a formulation of the Catalan number, in order to enumerate and catalogue all topological plan layout possibilities as the dual cell configurations of these triangulations in 2D;
- An algorithm for drawing weighted Voronoi (a.k.a. alpha shapes);
- An algorithm for constructing area-preserving 2D Isovist Bubbles; and
- An algorithm for constructing volume-preserving 3D Isovist Bubbles

The topological layout process in 2D is perhaps more interesting for urban layout process in which a 2D constraint is more often a case. It is fair to say that the architectural layout process introduced here is currently a theoretical investigation rather than being readily useable in architectural design practice. Acknowledging this fact, we see its contribution in two things: firstly in putting forward a new way of designing (i.e. through topological layout), that is purely configurative rather than figurative; secondly in provides a way of incorporating configurational analysis into parametric design workflows using a simple representation: bubble diagram.

§ 3.15 Future Work

Isovist Bubbles are goal-seeking agents that can interact and compete in order to reach their goal, i.e. their desired area/volume, while interacting with their fellow bubbles in 2D or 3D spatial containers. We have not yet implemented an algorithm for such Agent-Based Models but laid its foundation by introducing the agents. The prospect of using such models is that they can allow designing spatial layouts freely while preserving the desired configuration. In other words, they can give a second degree of intelligence to the interactive bubble diagrams drawn by the force-directed graph-drawing algorithm in that they can allow bubbles to grow into each other and deform each other.

The other area of this methodology that calls for future research is using the REL charts and From-To charts to attribute weights and directions to the nodes of a configuration based on the information in the charts. If we use the weights and directions as such, then the centralities computed on the configuration graph can be revised more consistently with the design intentions or requirements. In fact, the primary input of the design process can be based on REL charts and From-To charts.

In merging the two methodologies introduced in this book, we intend to replace/ extend our network analysis package by alternative models such as those of Chapter 5.

4 Implementation & Test A: SYNTACTIC

In this chapter, we introduce an implementation of the design methodology introduced in the previous chapter. The term methodology refers to a structured collection of methods, in the same way that the term technology means a structured collection of techniques. This is an important distinction: not every design technology can be regarded as a design methodology. Many design tools can be used in various situations without enforcing or suggesting a particular way of designing. A design methodology, on the contrary, suggests or enforces a certain way of designing or a way of thinking. The syntactic design methodology introduced in the previous chapter is suggesting a way of formulating spatial layout as a matter of configuration, i.e. focused on how spaces should connect to one another to serve best a functional program regarding such things as privacy, community, and frequency of movement through spaces or space usage potentials. The toolkit SYNTACTIC is made for computational design workflows and made available as a plugin application for the present-day de facto standard environment for computational design practice (Grasshopper3D). If this platform disappears, the toolkit can be adapted to another environment with relative ease. All methods have been implemented in .NET Object-Oriented Programming languages (VB.NET and C#.NET).

This chapter introduces the implementation of SYNTACTIC design methodology as follows:

- Expresses the goals and outlook of the toolkit as to its target users;
- Gives an overview of the guiding principles for its UI;
- Shows the whole structure of the design methodology;
- Introduces the tools in the toolkit individually;
- Showcases a few educational work samples;
- Concludes by providing a qualitative evaluation of the toolkit; and
- Discusses future development strategies for SYNTACTIC.

§ 4.1 Introducing SYNTACTIC: a toolkit for architectural configuration

The toolkit SYNTACTIC[a] (presented in Figure 50) is a plugin application consisting of tools for the graphical-algorithm-editor[b] Grasshopper3D©. Early versions of this toolkit were inspired by and compatible with SpiderWeb[c], developed by Richard Schaffranek, i.e. a toolkit for Grasshopper providing graph theory algorithms for configurational analysis and synthesis.

There have been several cycles of test and development for producing SYNTACTIC, out of which three versions have been released publicly as they were seen stable. The current version (2.7) is introduced in this chapter.

FIGURE 50 shows the logo and the appearance of the SYNTACTIC toolkit, version 2.7, as in Grasshopper3D© environment, version 9.0076, 2015; SpaceSyntax_for_GenerativeDesign by Pirouz Nourian & Samaneh Rezvani is licensed under a *Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License*. Based on a work at *https://sites.google.com/site/pirouznourian/syntactic-design*. Permissions beyond the scope of this license may be available at *http://www.grasshopper3d.com/group/space-syntax*.

a The toolkit SYNTACTIC has a dedicated user-group currently having 217 users worldwide: https://sites.google.com/site/pirouznourian/syntactic-design

b NOTE: the modules shown in this chapter and chapter 6 are made for Grasshopper3D, but they are made up neither of Grasshopper3D components, nor any of its plugins. We have not used any graph algorithm libraries either. The code behind the modules (the SYNTACTIC plugin) is written in VB.NET within Visual Studio. The environment of Grasshopper3D does not show the direction of data transmission in wires but it is always from left to right. Knowing the direction of wires, then a GH file is exactly a structured flowchart which does performs computation. However, the computation modules are all made up of Microsoft .NET Framework languages.

c http://www.grasshopper3d.com/group/spiderweb "SpiderWeb is currently a .NET library providing functionality for preforming calculations on graphs. The library is created with a special focuses on the integration of graphs and graph related theories (e.g. Space Syntax) in parametric design environments."

§ 4.2 Goals, Outlook and Target Users

The toolkit is primarily a manifestation of the best results of several experiments carried out in the direction of devising a robust configurative design methodology. The idea motivating the development process was that a methodology is only practicable given the right tools. Especially, if the methods are to be used in design practice -unlike many other research initiatives that have had only an academic life cycle- then the methods should be made available to anyone who might be interested in applying or testing them. Producing a toolkit that any designer can use with any kind of inputs requires a number of extra challenging steps to ensure simplicity, robustness, and user-friendliness. These challenges pay off when we receive crowd-sourced feedback on usability, and usefulness of the implemented methods.

The toolkit SYNTACTIC was envisaged as a primarily educational apparatus, which could gradually become more powerful by going open-source and thus transforming into a set of computational methods (functions with inputs and outputs) customizable to different workflows. The idea is that configurative design, as an approach, is not limited to any architectural style, structural system, or alike in that it is a primarily topological approach, which could be adjusted to be used in production of different geometries. The topological nature of the approach is exactly what makes it a potentially versatile methodology for architectural design.

Computational design has been being researched since 70's; we might even go back to 60's in its history; but it was mainly an academic hype until recently when environments such as Generative Components (GC)[a], Grasshopper3D (GH), Dynamo[b], and Design Script[c] came into existence.

It has been possible for years to practice computational design; and in fact, some firms such as Arup have used computational design approaches back in 70's. However, the use of computational design techniques used to be limited to large firms who could afford hiring programmers and specialists. The above-mentioned tools were in fact game-changers in that they made practicing computational design significantly easier, more manageable, more economically feasible, and more understandable for the majority of designers.

a www.generativecomponents.com/
b http://autodeskvasari.com/dynamo
c http://designscript.ning.com/

These platforms have had a paradigm-shifting role in making computational design significantly more available, feasible and much richer by virtue of bringing a huge number of practitioners and developers into the domain. In light of these new developments, we envision a paradigm shift in architectural education that has already manifested itself by the emergence of dedicated curricula for computational design. In this paradigm, the physical aspects of architecture such as geometry of environment and shapes, structural integrity, and climatic optimality have received weighty attention; however, focus on configurational aspects of design has been relatively less common. This is perhaps because of the relatively abstract nature of these issues. Nevertheless, 'Making spaces socially logical' should be at the heart of architectural education in that it is inherent in the commonsensical definition of architecture as a profession. We see the role of the toolkit as a facilitator of thinking for architects and architecture students in this direction, as it is now (a closed-source freeware). By going open-source in future, the toolkit can potentially spark a much larger movement in this direction.

As a progress-oriented strategy, we see many benefits in joining the two toolkits SYNTACTIC and CONFIGURBANIST (to be introduced in chapter 6) based on our [work-in-progress] computational library **configraphix.dll**[39]. This library can be used in developing different types of applications without even being dependant on an environment such as Rhino3D. Apart from technical advantages, such a unified approach to the configuration of built environments would bring about many methodological advantages as well.

§ 4.3 Designer-Computer Interface

We have had a number of **guiding principles** in implementing our design methodology as a toolkit regarding communication with the user (designer), in light of our general goals.

1. The method should demand as little information as possible and be as generic as possible; hence, no limiting assumptions about shapes should be enforced.
2. We want to free the initial steps of design from geometric constraints, so we will only ask about the spaces (nodes) and spatial connections (links), regardless of the geometry of them as points and lines.
3. We do not have any intention to make an automatic design algorithm; instead, we intend to reveal generic possibilities in an interactive manner and at the same time offer real-time analysis.
4. It is not for the system to judge the goodness of a solution, because we deem this matter a contextual intellectual task for the designer that cannot be formulated in a generic manner. Instead, we provide the designer with analyses and intuitive explanations as to how the Space Syntax measures can be interpreted. For example, it is not so that having spaces of high integration is 'generally' desirable; this is instead a contextual matter of fitting graph properties to what is needed in terms of privacy and community 'in a particular context'. However, as the measures are provided, an interested designer might come up with a formulation of a goal as such, and then use it for an automated evaluation procedure; but this would be an ad hoc scenario, not something to be generalized. Besides, the Space Syntax measures such as integration and control are completely neutral without a context.
5. The communication should occur in the order of abstract to concrete representations, such that abstract information has very little or no constraints.

§ 4.4 Design Workflow

The course of actions suggested by our proposed design methodology and supported by this tool suite is as technically described below (we have marked what is automated by the tools with bullets; the numbered items are what expected from the user to do with the tools)[40].

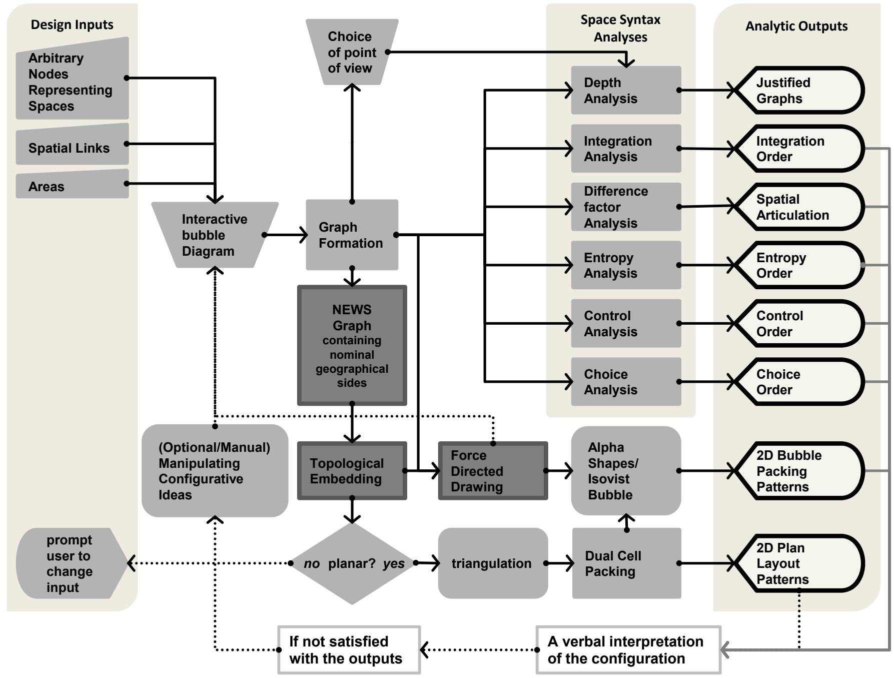


FIGURE 51 a flowchart describing our proposed design methodology. We could wrap the whole set of tools in this way, but we chose to let the curious user try different tool configurations.

The first sets of tools that a user needs are those shown in Figure 52, Figure 53, Figure 54, and Figure 55. The experimental or work-in-progress tools for geometric plan layout are shown in Figure 56. The following list describes the design workflow as User Experience (UX) that is suggested by the User Interface (UI) of the tools.

1. Start with putting a number of arbitrary points as for defining the centre of functional spaces
2. Optionally provide a list of (rough or exact)[a] area values for all spaces
3. Optionally provide a list of spatial labels (names) for the functional spaces
4. A tool assigns rainbow colours to the functional spaces to make them more recognizable (see Figure 53).
5. A method puts circles of sizes specified by the area values around all centre points, labelled automatically or as specified.
6. The graph reader tool provides a sketchpad with nominal "North, South, East, and West (NEWS)" sides for the user to draw the connections in (see Figure 55).

a If the user wants to achieve a needed total area, it is enough to provide rough values and the tool puts out a set of area values that exactly sum up to the required area.

7 According to your configurative ideas, draw a line between every pair of points (circles representing functional spaces) that you think should be directly linked. Add a few links to relate some of the spaces to the Northern, Southern, Eastern, or Western frontiers of the plan. These links could potentially guarantee that certain spaces be naturally lit in a desired way.

8 The graph reader interprets the input links and points and their "label & area & colour" attributes as a graph and provides the user with a verbal interpretation of links between spaces.

9 The Bubble Diagram tool (Force-Directed Graph Drawing Algorithm) shows a neat drawing of the architectural configuration graph.

10 Connect the Space Syntax real-time analysis tools and reflect on the analyses; see if your initial design intentions (in terms of such things as privacy, community, popularity, frequency of usage, ease of navigation and alike) are actually satisfied within your configuration. It is for you to interpret the results and change the input if you find that necessary.

11 Optionally draw the same connectivity pattern using the NEWS graph tool, while adding enough connections to the nominal quadruple sides; then use the Tutte Convex Drawing tool to obtain an embedding of your configuration graph. This tool untangles the proposed bubble diagram and delivers a planar convex drawing of the connectivity graph[a]; and distinguishes a sub graph of the whole connectivity graph (excluding NEWS vertices). You can now find all possible triangulations of this embedding, each of which corresponds to a distinct plan layout topology for your configuration graph.

12 Out of the plan layout topological embeddings acquired before, you can try to find plan layout geometries using the next set of tools, out of which the alpha shapes is standardized and stable. The rest of the tools are either experimental or in a work-in-progress status. You can also develop your own tools to convert a topological layout to a geometric layout.

Note that an ad hoc process for converting a plan-layout topological embedding to a geometric plan can be much less sophisticated than a generic-purpose systematic process. We have sparked a research-oriented design process by providing **Isovist Bubbles**. These computational geometry constructs can potentially elevate the sophistication of the plan layout process and provide for a very generic process that can be applied to all kinds of boundary shapes to ensure a certain plan topology while realizing spaces as specified in terms of areas, guaranteeing that they remain as star-convex or isovist shapes.

a This is possible only if the graph is planar per se; in other words, if a planar embedding of the graph exists, it will be the output of this algorithm, meaning that if the output is not planar, the input graph is not planar.

§ 4.5 Tools

The toolkit[41] provides 5+1 groups of tools as shown in Figure 53, Figure 54, Figure 55, Figure 56, Figure 57, Figure 58, and Figure 59. The sixth group, i.e. the Isovist Bubbles, have not yet been released publicly. The standard parameter components used for getting inputs from the user are shown below. The advantage of having labels is apparent but they are optional, area values would be necessary in approaching a geometric layout. The total area can be inserted to be enforced as a total sum; meaning that, if the sum of areas does not meet the exact area available, we can regard them as portion weights and ensure the sum to be the specified amount.

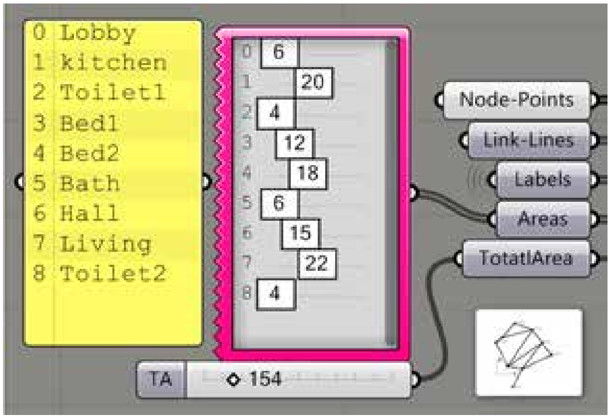


FIGURE 52 shows the initial configurative inputs from the user (designer). The essential inputs for constructing a configuration graph are points and lines.

§ 4.5.1 Graph Formation and Graph Drawing (Figure 53)

Among this set of tools, the one that requires most interaction with the user is the Node-Link graph formation tool. As explained in previous chapter, this tool must construct the graph based on a set of points and a set of lines. A reasonable sequence is to put in points first and lines next. Once the user puts in the points, the tool automatically assigns cardinal numbers to them and labels them as such in the geometry environment of Rhino3D®. Optionally, the user can put a corresponding list of labels for the nodes referring to their actual names. In addition, area values can be put in for forming a bubble diagram. If the use does not provide such values, the tool automatically assigns monotone values based on the relative size of the whole configuration as drawn by the user. This tool also assigns a range of colours from a rainbow spectrum to help the user more easily distinguish the spaces from one another. Only the node and link inputs at the left side are mandatory, and attributes are optional. Our force-directed graph-drawing algorithm makes a neat bubble diagram quickly for small graphs, if the graph is planar and that the combination of areas allows for a coin-graph drawing.

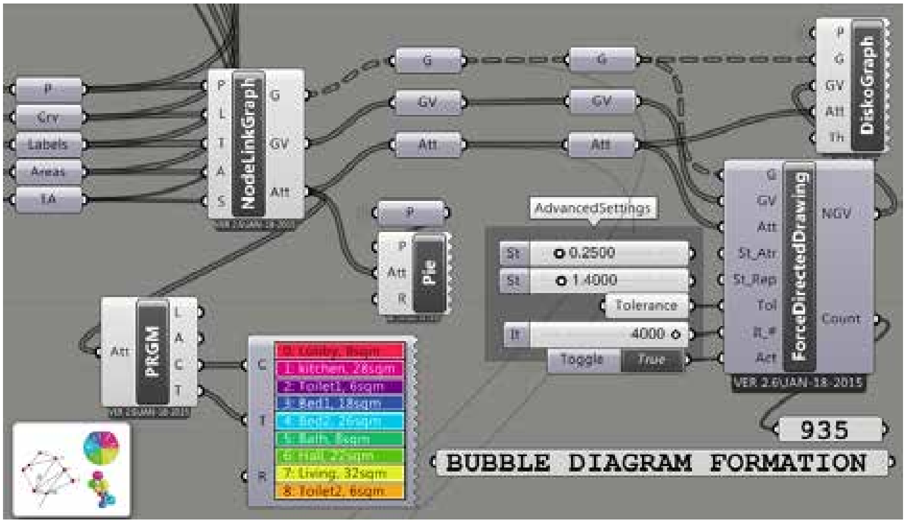


FIGURE 53 shows the first group of tools from SYNTACTIC for producing a graph from points and lines, Force-Directed graph-drawing algorithm, pie-chart drawing and attribute set composition tools.

§ 4.5.2 Space Syntax Analysis (Figure 54)

As came in the chapter3, we have implemented a set of tools for performing Space Syntax analyses in real-time. What is meant by real-time is the fact that firstly, these analyses run quickly and simply by receiving a graph input, which can be from any computational process. Secondly, the tools can be immediately uses as soon as a designer puts in a graph composed of nodes and links, i.e. the simplest or the most abstract form of a spatial idea. Each of the tools also features short intuitive introduction of each measure in line with the descriptions given in the previous chapter. The measures have been **verified** comparing with those of DepthMap for small plans.

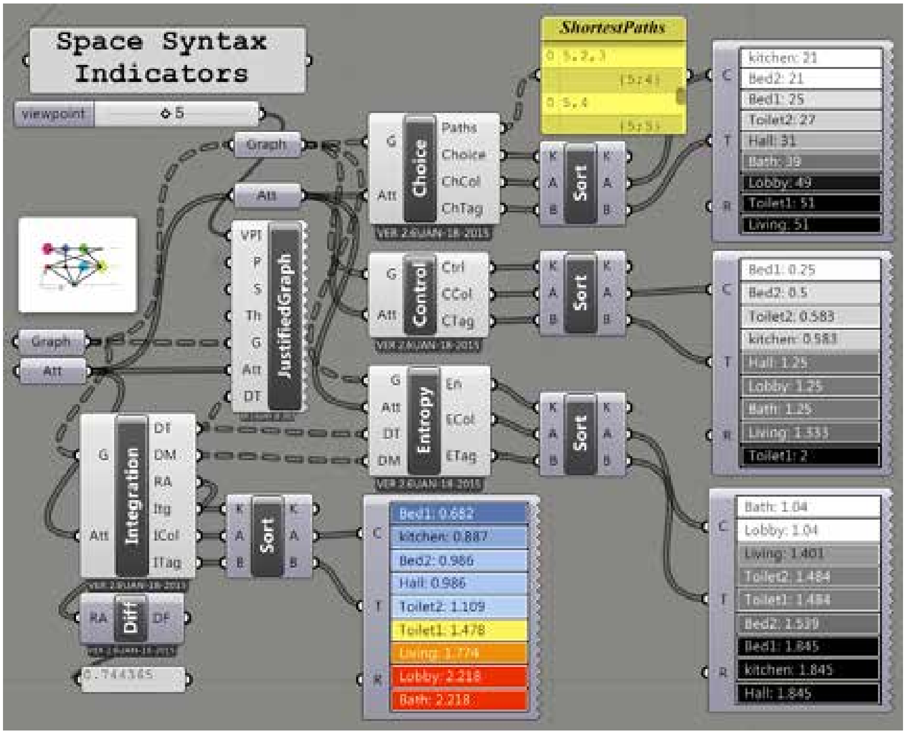


FIGURE 54 shows a group of tools from SYNTACTIC performing Space Syntax analyses and reporting orders of Space Syntax measure on a sample architectural design assignment. Our Justified Graph component provides a unique opportunity for designers to draw justified graphs easily from different viewpoints.

§ 4.5.3 Topological Embedding for Plan Layout (Figure 55)

This set of tools help a designer reveal the set of possible planar topologies for plans, each of which admits the connectivity graph put in by nodes and links. The NEWS-Graph tool produces a virtual sketchpad for the user in the geometry environment to guide them in fixing connections to the four nominal sides of the plan in order to derive at the unique convex topological embedding from the Tutte algorithm tailored for this purpose. Based on a convex embedding delivered by Tutte algorithm we find all possible maximal planar triangulations, each of which corresponds to a dual plan topology.

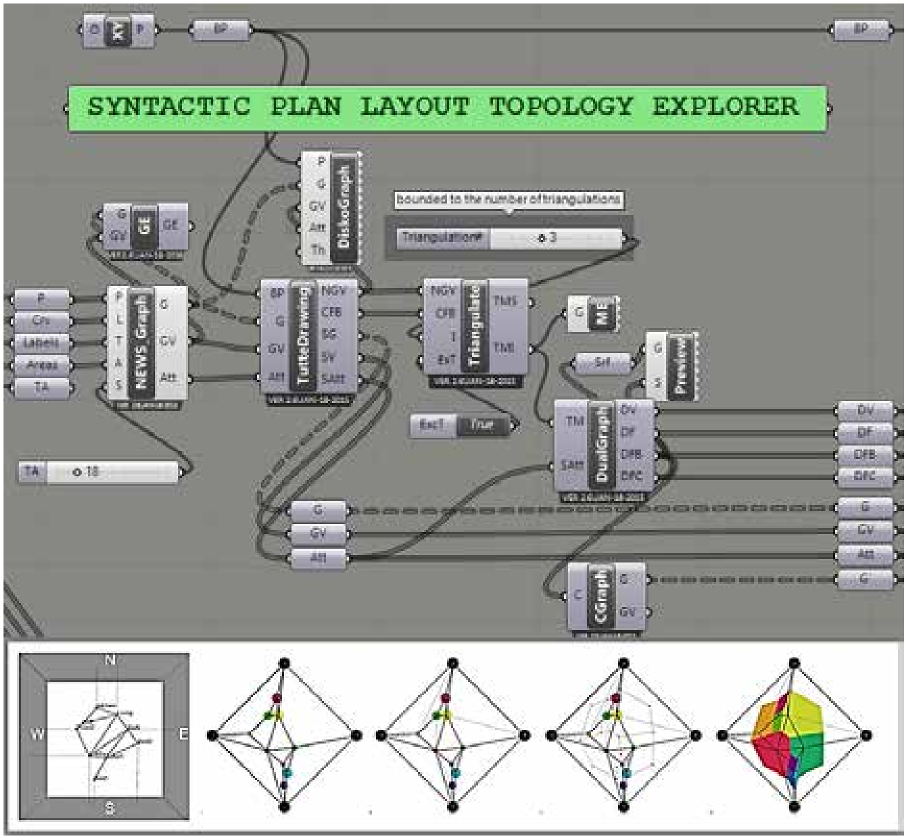


FIGURE 55 shows SYNTACTIC plan layout topology tools and their typical outputs.

§ 4.5.4 Geometric Plan Layout (Figure 56)

A key tool in this set of tools is experimental and unstable: the Rectangulate tool only mimics the process of rectangular graph drawing by physically enforcing a plan layout topological embedding to become rectangular by means of recursive algorithms. This is not a standard or stable process for this purpose; nevertheless, it has been provided to show case a number of possible directions for concretization of layout topologies as layout geometries.

The tool called dimensionless provides an algorithm for drawing "dimensionless rectangular dissections" as defined by Steadman (Steadman, 1983). This tool, independently from the rectangular drawing tool, can convert a rectangular dissection (rectangular cell division) into a topologically homoeomorphic (say equivalent) dissection whose cells have side lengths as multiples of an arbitrary size. In other words, a rectangular cell in a dimensionless dissection can always be constructed out of multiple squares joint together. An example of such a drawing is apparent in Figure 56. Such dimensionless plan layout patterns can be later dimensioned using an approach based on Kirchhoff circuit laws, as described in (March, L, Steadman, P, 1974), (Roth, J & Hashimshony, R, 1988), and (Steadman, 1983) . As explained in the previous chapter we did not follow this line of work, for we preferred to focus on more-generic processes that do not impose a certain shape such as rectangles to the concretization process.

The Alpha Shape component provides alpha-shapes using an algorithm based on the process defined in (Edelsbrunner, H., & Harer, J., 2008). This component is supposed to give a tangible idea of colliding spaces as bubbles (e.g. by pushing the bubbles of a bubble diagram towards each other) into a confining boundary shape. It does not satisfy the ultimate goal of packing bubbles as room representatives as it fails to maintain the area of bubbles. This is the motivation behind development of Isovist bubbles, which necessitate the implementation of an Agent Based Model.

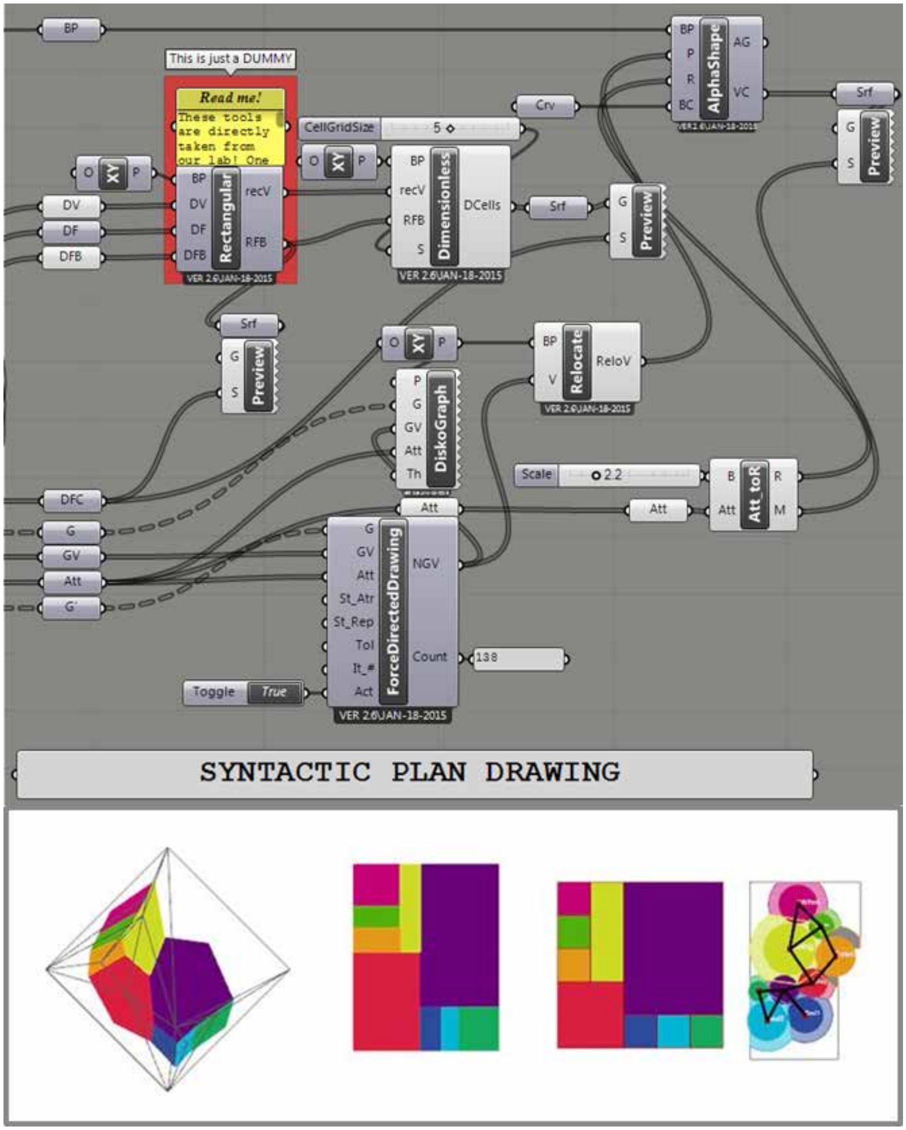


FIGURE 56 the set of tools from SYNTACTIC for making geometric plan layout patterns from the catalogue of topologic possibilites revealed by the preceding set of tools. The rectanuglar drawing tool is only a place holder at the moment. It uses a heuristic instead of the algorithms based on Kirchhoff Laws.

§ 4.5.5 Isovist Bubbles for an Agent-Based Model for Spatial Layout (Figure 57 & Figure 58)

Isovist Bubbles, as introduced in the previous chapter are produced as smart agents for Agent-Based Models for spatial layout. Their own complex nature and the intricacy of their implementation has slowed down the process of development of an ABM but the two agents IVB2D and IVB3D are working robustly at the moment. IVB2D is implemented in VB.NET and IVB3D is implemented in C#.NET.

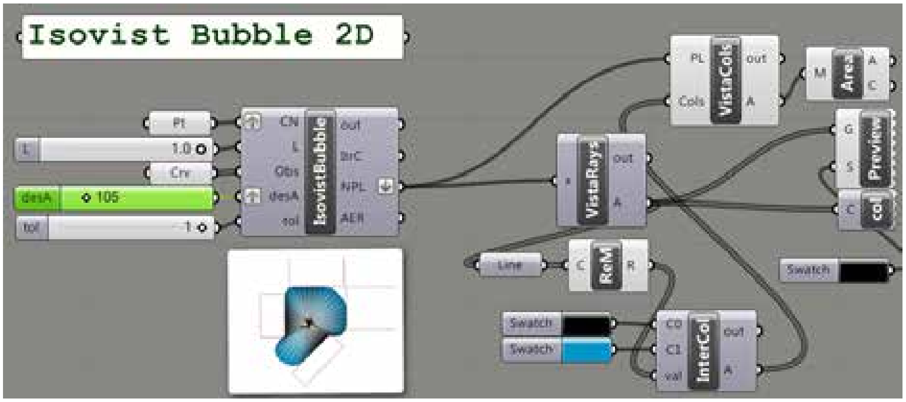


FIGURE 57 the IVB 2D tool (written in VB.NET) for SYNTACTIC and some related utility tools (WIP).

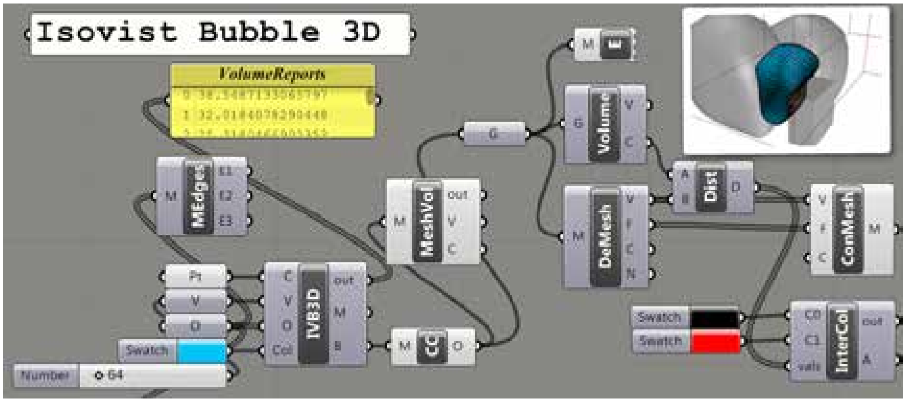


FIGURE 58 the IVB 3D tool (written in C#.NET) for SYNTACTIC and some related utility tools (WIP).

§ 4.5.6 Urban Configuration Analysis Tools (Figure 59)

Axial Line Graph tool (abbreviated as ALG) and Local-Integration tool are made particularly for urban configuration analysis, but the rest of the tools need only efficiency improvements and implementation of lighter data structures to be used in an urban scale, as their algorithms are the same as those used in architectural configuration analysis. As stated in explaining our goals and outlook for the future of the toolkit SYNTACTIC, it is intended to bring this toolkit together with our urban configuration analysis toolkit CONFIGURBANIST into a single package by implementing and publishing a single library called **configraphix.dll**. It can be seen here why this move would make sense as to the similarity of methods and data structures used in both cases. We shall discuss this issue further in chapter 6.

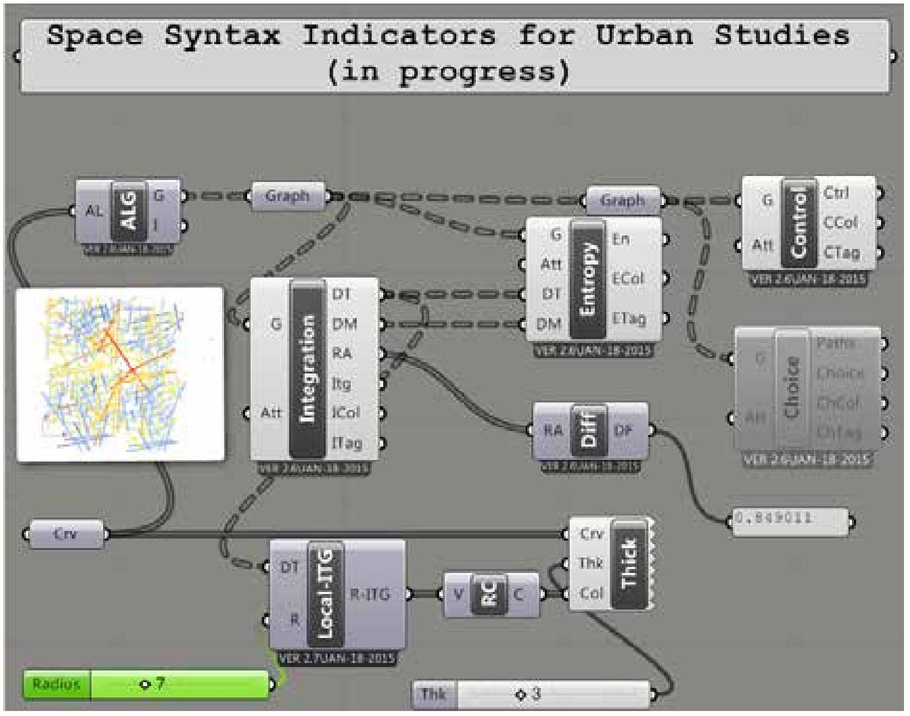


FIGURE 59 shows SYNTACTIC tools for urban configuration analysis.

§ 4.6 Educational Use

The toolkit SYNTACTIC has been offered to MSc architecture students in two 12 ECTS courses, respectively called XXL Design Workshop and High-rise Design Workshop at TU Delft, where the author has been involved as responsible instructor of computational design. The course offers an intensive design workshop for eight weeks, in which multi-disciplinary teams of students are supposed to work together, as if in a firm, to deliver a project that is challenging in terms of size (horizontal span or height, thereby the names XXL and High-rise) and complexity. The following work sample shows processes in which SYNTACTIC has played a role. It is an Indoor Ski Resort, Computational Designer: Rusne Sileryte, Structural Designer: Zejun Pei, Climatic Designer: Vivian Wijburg, Architectural Designer: Joost van de Ven.

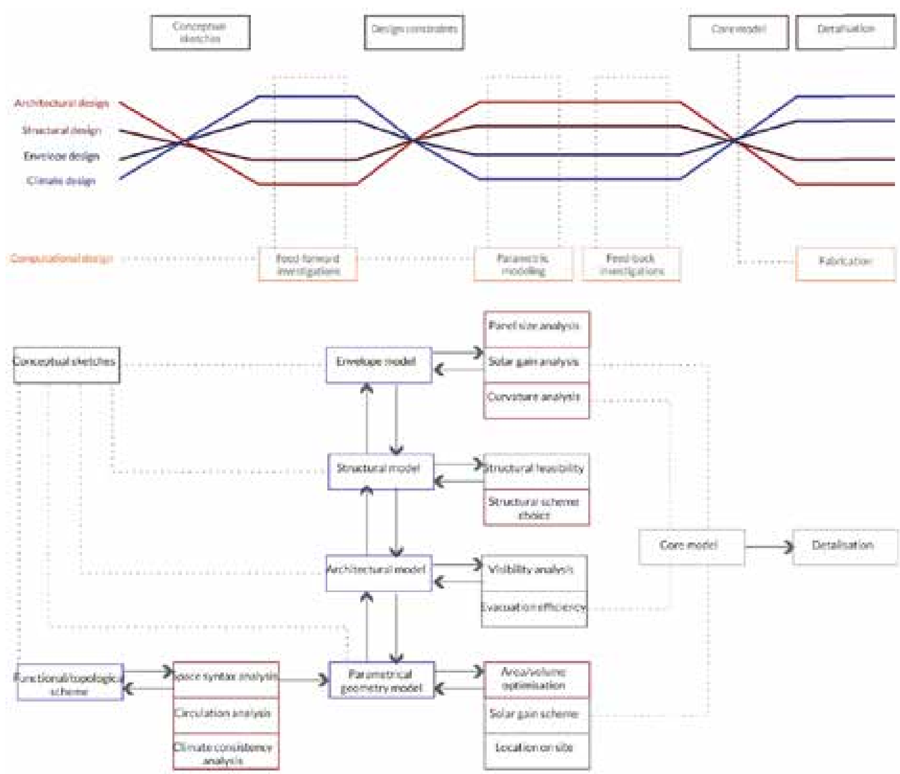


FIGURE 60 a phase model and a block-diagram view of the multi-disciplinary process. Image courtesy of Rusne Sileryte.

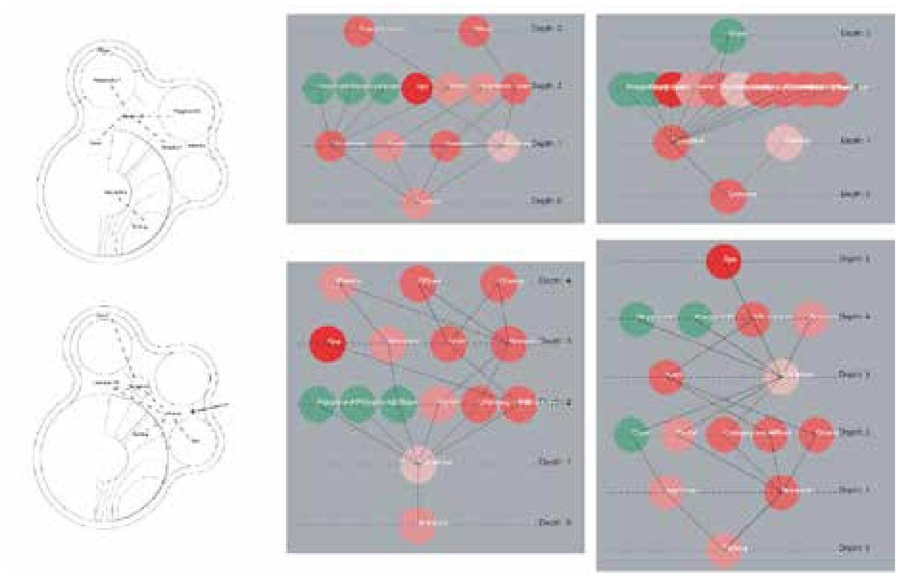


FIGURE 61 shows snapshots of the process through which the initial connectivity pattern was chosen. Note that the circles come from the architectural-structural concept, not from the bubble diagram tool. Team members brought their ideas on how the spaces should connect to one another; they colour-coded the spaces as to their temperature, green representing cold and red representing warm; drawing their connectivity patterns using the Node-Link Graph and Justified Graph tools of SYNTACTIC they could explicitly negotiate the pros and cons of each connectivity pattern. Image courtesy of Rusne Sileryte.

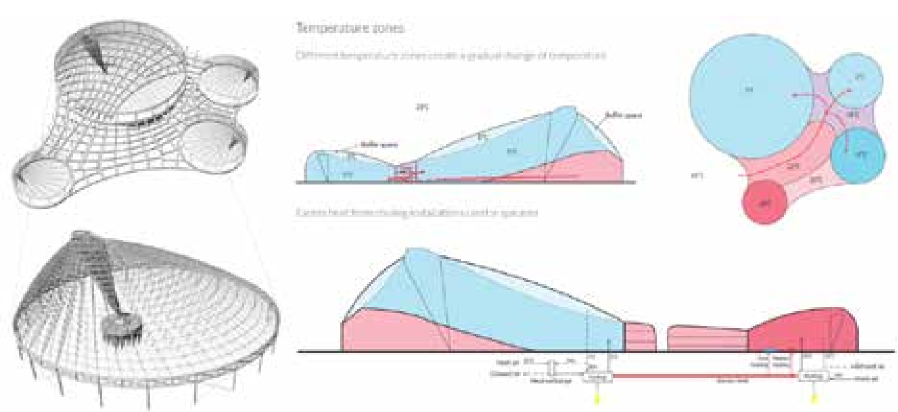


FIGURE 62 shows the structural system and the climatic arrangement of spaces; images courtesy of Zejun Pei and Vivian Wijburg.

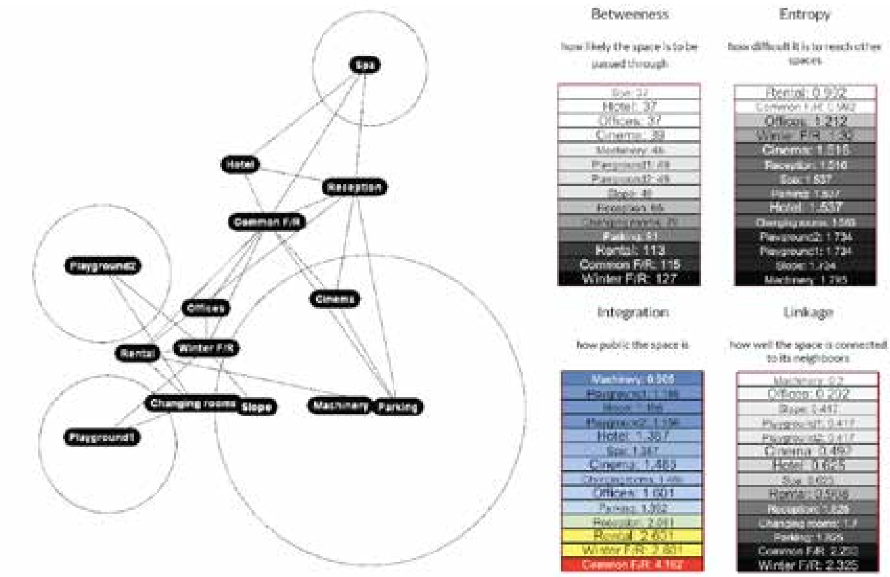


FIGURE 63 shows the Space Syntax analyses done using SYNTACTIC to finalize a connectivity structure. Proposed configurations were edited to achieve reasonable spectra of privacy and publicity (integration), frequency of passage (betweenness), ease of way-finding (entropy), and connectivity (control). Image courtesy of Rusne Sileryte.

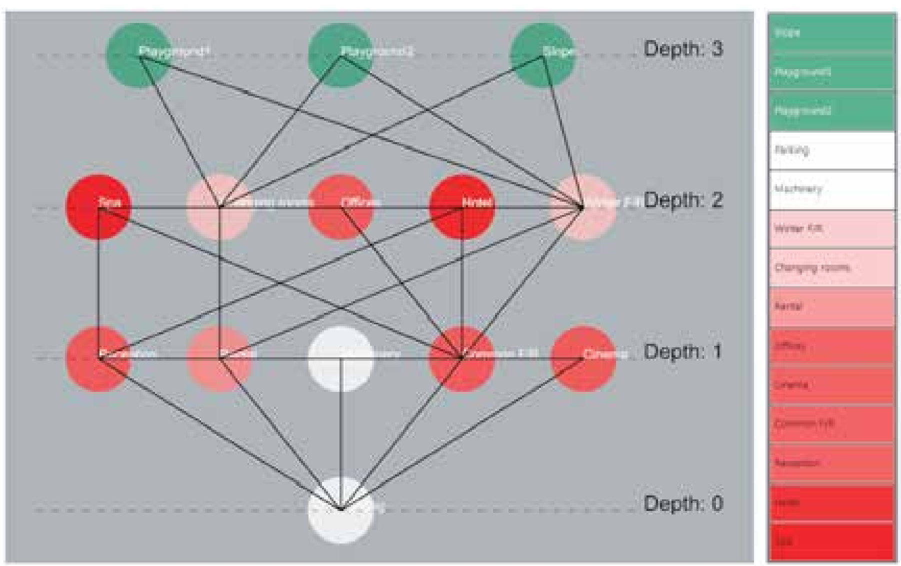


FIGURE 64 the final connectivity graph drawn using the Justified Graph Drawing tool and coloured as to temperature. Note that the team managed to make a match between the climatic spectrums of cold to warm spaces -as was needed for health and sustainability reasons- and the connectivity pattern in terms of the right sequence of access for different spaces as to their required level of public exposure. Image courtesy of Rusne Sileryte.

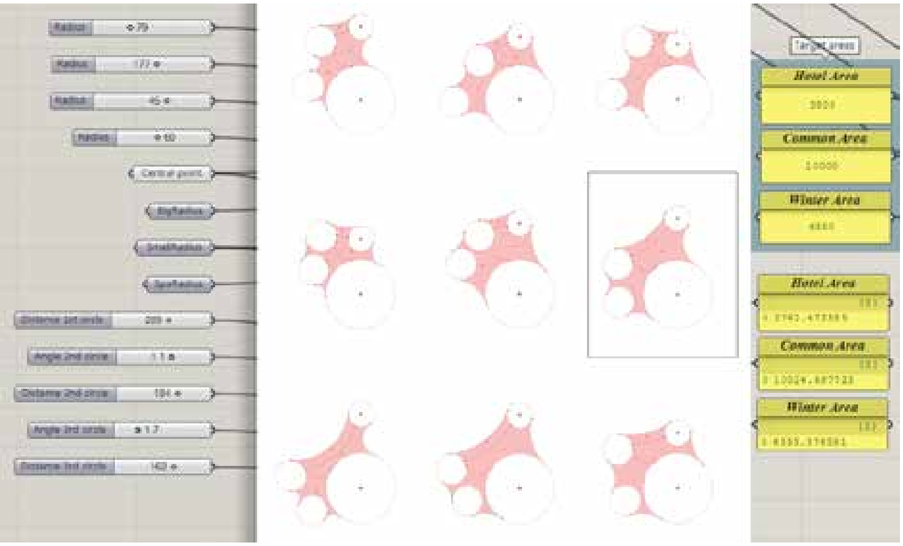


FIGURE 65 following the connectivity pattern chosen as final, a cell configuration topology was decided for the space in between the circular structures. Three main regions were designated for a hotel, a common area and a winter area, the adjacency of these cells to themselves and to the circles had to be preserved while getting as close as possible to the values from the physical programme. A custom-made optimization loop was made to achieve this end. The figure shows the converged solution for this problem and the optimum plan geometry given the formulation of the form and the probelm. Image courtesy of Rusne Sileryte.

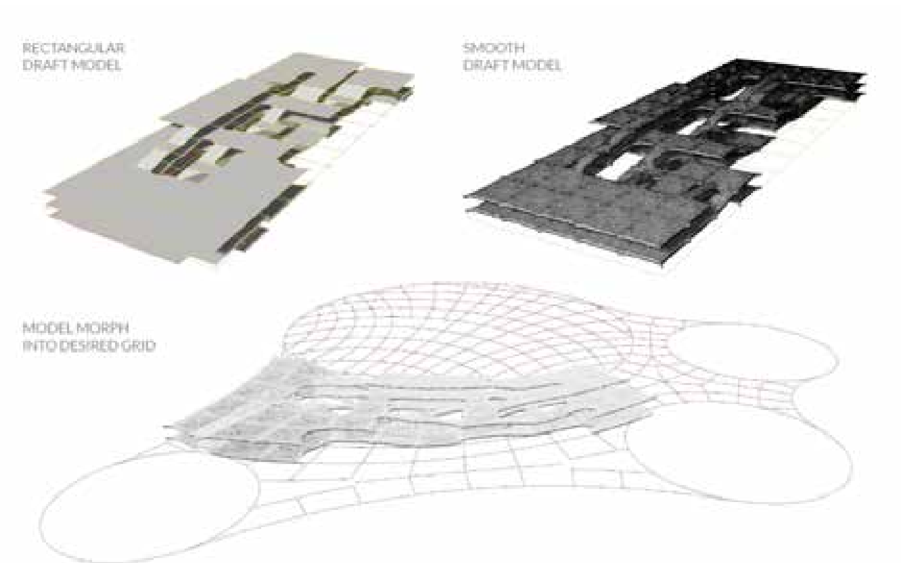


FIGURE 66 shows a circulation system design for the main part of the spaces in between the circles; following the configurative approach, circulation was first designed as a rectangular form, then tessellated as a mesh and finally morphed into the actual shape. The computational designer was instructed to design topologically and morph the simple shape into its homeomorphic curvy shape. Image courtesy of Rusne Sileryte.

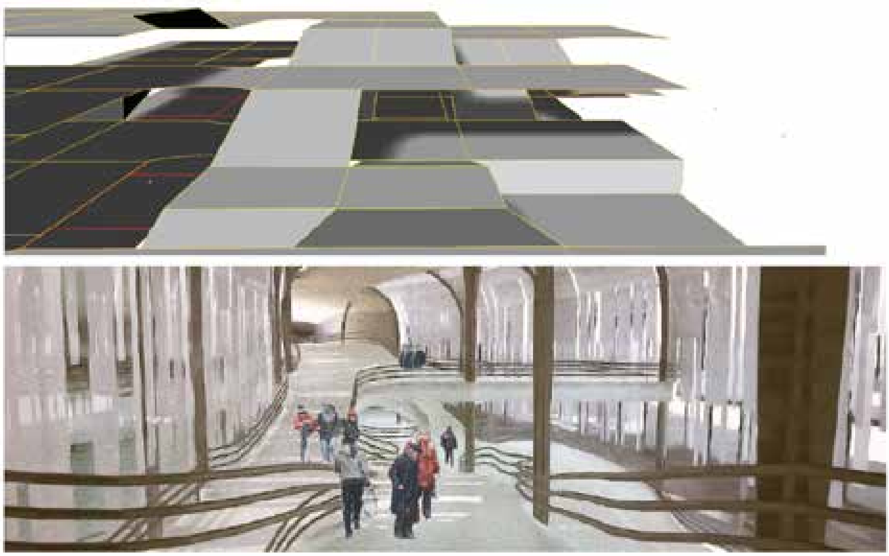


FIGURE 67 shows a closer-up of the circulation structure and its image when rendered as in final curvy shape. Image courtesy of Rusne Sileryte.

FIGURE 68 shows a closer-up of the circulation structure and its image when rendered as in final curvy shape. Designing the final shape directly would have been very difficult, using the topological approach instructed this complex shape was achieved systematically according to the well-thought spatial configuration. Image courtesy of Rusne Sileryte.

Other examples might be found on the user-group of SYNTACTIC.

§ 4.7 Achievements and Limitations

It appears from its public user-group that the toolkit SYNTACTIC is being used by students and computational design practitioners worldwide. It has certainly helped in making Space Syntax analysis easily applicable in computational design workflows and exemplified the idea of integrated configurational analysis and synthesis. Implementing SYNTATCIC as a toolkit and making it publicly available as freeware has succeeded in bringing "configurative design" thinking to the centre of attention in the computational design community. Other toolkits offering Spatial Network Analysis tools, applicable in parametric architectural design processes are available for computational design practitioners, prominent examples include:

- **Spiderweb**[42], by Ir. Richard Schaffrane at TU Wien
- **Decoding Spaces**[43], developed by Dr. Reinhard König at ETH Zurich; Dipl.-Ing. Sven Schneider at Bauhaus Weimar, and Ing. Arch Martin Bielik also at Bauhaus Weimar
- a live connection between **DepthMap to Grasshopper**[44], by Dr. Tasos Varoudis at UCL

As mentioned above, the difference of our methodology with these examples is that it is intended to provide **a way of designing with configuration**, not merely spatial analysis. However, it can be said that this way of designing is yet more of a theoretical investigation into the fundamentals of configurative design. In other words, in its current form, it will be difficult to use in design practice. SYNTACTIC has not yet achieved the ideal of making design process fully configurative, i.e. starting from a configuration and arriving at a form considering constraints and performance criteria. Literally achieving this goal requires many more cycles of contemplation, design, development, test, and verification. Reaching to this very point has been quite challenging in terms of implementation effort. A very challenging aspect of the development is the complexity of User Experience (UX) and User Interface (UI) required for providing a designer-friendly software application.

The toolkit has been seemingly well-received by its international users despite the facts that: it enforces using a relatively strict design methodology; and that it deals with the most abstract aspects of architectural computation, as compared to climatic, formal, or structural aspects.

SYNTACTIC does not offer very straightforward indications on goodness of configurations; instead, it encourages designers to think of and deal with configuration and its likely social effects explicitly in their design process. The analytic engine implemented in the toolkit is a set of Space Syntax measures, adapted and interfaced so that they can be used in computational design workflows. However, interpreting the meaning of these indicators is often a challenge for the users of the methodology.

The Space Syntax analytic engine can be considered as a starter, i.e. the best option available by the time of implementation, as an established Spatial Network Analysis theory that has been validated to be useful in studying spatial behaviours of humans in architecture. It can be seen that a revision or re-interpretation of such measures would be helpful in order to make them more directly useable in design practice. In other words, more readily understandable measures such as probabilistic measures (e.g. those introduced in Chapter 5), could be potentially better applied in design thinking.

The synthetic part of the methodology is the one that is most complicated and intricate. If we did not offer a clear configuration synthesis workflow, then the analytic engine would have been rather banal. To promote a new way of designing or a new "designerly way of thinking", we needed to provide a new way to synthesize form. This is inevitable; and yet, without adopting a strategy to limit the possibilities for topological embedding, the task of embedding a graph in Euclidean space would become too difficult to handle or to be even understood by a typical user. The path that we have followed for embedding plans in 2D has successfully ended in enumerating all possible plan layout topologies; but it is still very limiting for real-world architectural design. It is perhaps more interesting for urban design explorations, where adopting a planarity constraint is often necessary. Anyhow, the future of the SYNTACTIC methodology could be determined by feasibility and applicability of the ultimately unrestricted configuration process with Isovist Bubbles achieves.

§ 4.8 Future Work

We have not yet achieved the ultimate goal of proposing a completely generic configurative process, which could lead to concrete geometric forms. However, we envision that if such a goal is realizable, a reasonable way to go for this goal is through Agent-Based Models and the application of Isovist Bubbles. The prospect of future developments of SYNTACTIC in the design direction is to advance this approach. On the other hand, the spatial analytic package should be extended and enriched with alternative spatial network measures, particularly those pertained to Markov chains and Eigen systems.

In determining future work directions, we must note the difference between the methods that guarantee a solution and those that are used conventionally because of their good results, i.e. heuristics. An example is the force-directed graph-drawing algorithm; it is not guaranteed to work under any circumstances, and there is no proof of its convergence; however, it is very popular because it is intuitive and that it works most of the times! On the other hand, the Tutte algorithm for graph drawing is guaranteed to work under certain circumstances but it does not always produce good[45] results.

Embedding a graph on a 2D plane and finding all possible cell configurations to admit connectivity graph was already a huge challenge in this work, which was resolved mathematically by methods that are guaranteed to work under certain conditions. It would be indeed very desirable to generalize such a process to 3D, but "3D is not just about 1D +2D; it is much more than that"[46]. To give the reader an idea of the complicacy of the process, we can mention that the problem of finding all possible plan layouts stayed unsolved for about a year. It was about finding all triangulations of a map composed of convex polygons. A similar problem in 3D would have been about finding all possible tertahedralizations of a cell configuration composed of 3D convex polyhedrons. It is obvious that the latter case is significantly more challenging and complicated, let alone the number of possibilities that grows much more rapidly in 3D. On the other hand, an Agent-Based Model with Isovist Bubbles could become a workable solution like force-directed graph drawing algorithm but it would be very difficult or probably impossible to prove its convergence mathematically. Although this sounds like a dilemma, practically, the way to go seems to be about moving in both directions: empowering the theory behind spatial configuration synthesis with strong mathematical models while, at the same time, providing methods (e.g. heuristics) that can be used easily in practice.

5 Model and Methodology B: Urban Configuration

This chapter introduces a methodology for computational analysis of urban configurations in terms of walking and cycling mobility and accessibility. Specifically, we introduce a path-finding algorithm for walking and cycling, a method for computing actual walking and cycling distances, a fuzzy definition of accessibility, a few probabilistic measures of passage of pedestrians and cyclists. This chapter[a]:

- Begins by explaining the societal relevance of the work;
- Clarifies the methodological point of view;
- Defines urban configuration formally;
- Introduces alternative mathematical models of spatial networks;
- Introduces methods for finding best paths for walking and cycling;
- Provides a Fuzzy Logics framework for measuring accessibility;
- Revisits a few centrality measures and reintroduces them according to the Fuzzy Logics framework laid out before;
- Introduces probabilistic models of passage of pedestrians and cyclists using Eigen systems, Markov Chains, Random Walks, and Google Page Rank;
- Summarizes the achievements and limitations; and,
- Concludes by discussing future work plans

Goal: To deliver a methodology for analysing the effects of spatial configuration on walkability and bikeability

Question: How can we model the effect of spatial configuration on accessibility (e.g. by walking and cycling) and mobility potentials? (Chapters 2, 5, 6)

a This chapter partly reflects points from two papers and a book chapter previously published, in which the author is the first co-author: (Nourian, P, Rezvani, S, Sariyildiz, S, van der Hoeven, F, 2015), (Nourian, P, van der Hoeven, F, Rezvani, S, Sariyildiz, S, 2015), (Nourian, P, van der Hoeven, F, Rezvani, S, 2015)

§ 5.1 Motivation, Societal Relevance and Scientific Relevance

It might sound apparent that promoting walking and cycling as green and healthy alternatives for urban mobility is desirable. Nevertheless, apart from good intentions, we would like to discuss the specifics of these modes of transportation and their relation to the form and configuration of built environments. We want to see how exactly the shape of a city and its geographic specifications might affect its potentials for walking and cycling. In urban planning and urban design, the actions with which improvements can be sought are not limited to spatial actions. It is true that the role of policy works, cultural campaigns, and other such actions might be as important as the role of physical interventions in the built environment. However, here we focus on the exact physical effects of the shape of built environments on actual distance, mobility, and accessibility. In other words, our focal area is spatial planning and policies that deal directly or indirectly with physical intervention and facility designation in cities, or management of the built environment.

The research reported in this chapter is to offer new methods, techniques, and tools for studying the relation between built environment's form and active mobility. We will not arrive at the ultimate goal of adding to the body of empirical knowledge on such matters; instead, we offer new analytic means and vistas that can be used by prospective researchers in that direction.

It is clear that before intervening in the functioning of a complex system one must have a reliable understanding of its current functioning dynamics and a diagnosis of problems to be solved by an intervention. In the case of mobility and promoting alternative modes of mobility, we propose that an analytic insight on the relation between the static form and configuration of a city and its dynamics is necessary. Evidently, such relations exist. However, in urban studies we need to acknowledge the multiplicity of causes or reasons behind phenomena.

It is not so that if the built environment is very appropriate for walking and cycling mobility people would necessarily adopt them as their primary modes of transportation; but of course, the 'potential' of an environment for certain interactions is of primary importance. It is such potentials that we want to measure and model in this chapter; admitting that there will always be differences between the potential and the actual. If a model can provide reliable insight on the potential then it is a good model. Note that in the quantitative sense the best model is the model that can give the best prediction about the future of a system, while being easy to use, requiring not too many data inputs. However, qualitatively, a good model is the one that gives the clearest insight on the way the system works. As an example, we can imagine an Artificial Neural Network can be potentially trained well in order to predict the near future states of a system, say the number of people walking through streets of a neighbourhood. However, such a model would give little or no insight on how the whole system works; it could just mimic the whole system in terms of its dynamics. On the other hand, a graph theoretical model can show us how exactly the structure of the system can affect the choices of the individuals and thus 'lead' the dynamics of the system.

The slightest change in the built environment's shape and configuration could have an effect on the functioning of the system as a whole. The question is on how we can gain insight on such effects; even if the relation is not exactly of cause-effect type, the configuration of environment can affect the choices of individuals and thus indirectly lead to patterns of 'collective behaviour'. Should we just go for interventions based on such things as intuition, political will, or expert opinions? This is how many such decisions are made in planning practice currently. Alternative methods could be knowledge-based and evidence-based, supported by analytic tools. This is the essence of what we are proposing: a spatial planning approach that is based on analytic tools that can help planners see what would be the likely effect of their plans in the mobility of people. At the core of such an approach, there should be analytic engines to model urban networks and their functioning in terms of likely collective patterns of spatial behaviour of people.

During the 1950s, prioritising vehicular traffic (and the private motor car in particular) emerged as a new trend in urban planning. This prioritisation manifested itself in a variety of ways: the planning of hierarchical networks of streets and roads (often based on the 'predict and provide' principle), the widening of streets and roads and the related demolishing of buildings and entire neighbourhoods, granting priority to car traffic, and turning former public spaces into arterial roads and car parking facilities. Such interventions have placed the needs of cars and other motor vehicles above those of pedestrians and cyclists, and created cities that are fragmented by roads. Additionally, rising car-ownership levels coupled with the priority given to cars by municipal authorities have made many urban regions car-dependent and forwarded urban sprawl, i.e. a well-known environmental problem. As a result, "cities suffer most from congestion, poor air quality and noise exposure. Urban transport is responsible for about a quarter of CO_2 emissions from transport, and 69% of road accidents occur in cities" (European Commission, 2011, p. 7)(Section 2.4 article 30). The European Union recognises the great potential of walking and cycling for reducing dependence on motor vehicles: "In cities, switching to cleaner transport is facilitated by the lower requirements for vehicle range and higher population density. Demand management and land-use planning can lower traffic volumes… Facilitating walking and cycling should become an integral part of urban mobility and infrastructure design, (European Commission, 2011, p. 7) (Section 2.4, article 30 &31)".[a] See also (European Commission, 2007a).

a From BIPEDALISM research proposal for the EU Horizon 2020 grant application for the call MG 5.3 2014, developed together with partners from UCL (UK), PBL (NL), Space Syntax Ltd (UK), Transport Insights Ltd (IE), Open

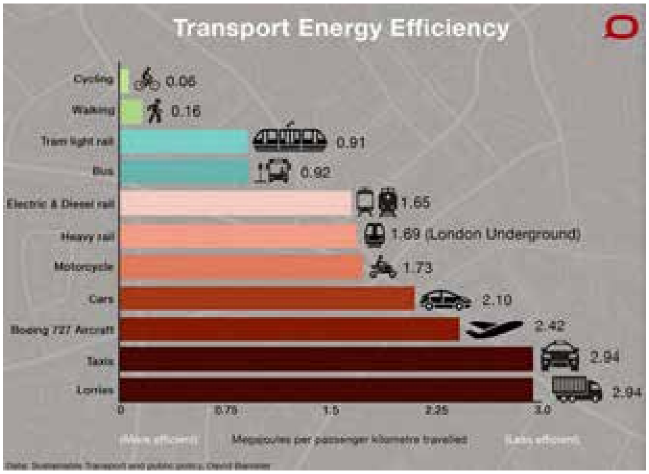


FIGURE 69 Data from (Banister, 2004), Image Credit: Tree Hugger (http://www.treehugger.com/bikes/trying-travel-city-bikes-are-most-efficient-way-move.html)

Despite the new trend and public interest in promotion of walking and cycling as sustainable modes of transportation, there is relatively little known about the dynamics of walking and cycling mobility as compared to vehicular traffic.

We believe that analytic insight into walking and cycling behaviour is needed in developing new infrastructure as well as adapting existing infrastructure to provide more people with better walking/cycling access to amenities. A few spatial network analysis models (such as Space Syntax) describe cognitive aspects of way finding for pedestrians and cyclists and there is a conventional approach of using shortest paths in modelling transport (Dios Ortuzar, J., & Willumsen, L. G., 2011, p. 358). However, conventional shortest paths[47] found on 2D maps are not very useful in studying walking and cycling because they cannot incorporate the physical, cognitive, or social aspects of wayfinding for walking and cycling. Pedestrians and especially cyclists, tend to avoid steep routes for their dependency on their own limited power, while this is not an issue in vehicular traffic modelling and that is why shortest paths are widely used in modelling the spatial dynamics of vehicular traffic. We assume that journeys made by bike or on foot need to be easy (physically and cognitively), safe and socially pleasant; and argue that we need alternative analytic methods to address these issues.

We intend to provide a framework for studying walking and cycling mobility to reveal the relation between the shape and configuration of built environment with accessibility and mode-choice behaviour. The core of this framework is an optimal path (geodesic) algorithm that could show how an individual could choose a path that is

Sky Data Ltd (IE), Samenwerking stadsregio Eindhoven (NL), Câmara Municipal de Lisboa (PT), Municipality of Ljubljana (SI), and ECTP-CEU [European Council of Spatial Planners - Conseil Européen des Urbanistes] (BE) relatively easy for travelling from an origin to a destination considering the particular physical, cognitive, and social aspects of walking and cycling. As an ultimate goal, we envisage developing an analytic engine that can form the basis for a Planning Support System (PSS), which can be used in planning cycling infrastructure and developing pedestrian & cyclist friendly neighbourhoods.

When modelling vehicular traffic, we do not require reflecting the physical difficulty of going uphill. If one is driving a car, then one just presses the gas pedal and goes ahead. Driving does not necessarily have to be very intuitive either, one is guided by a system of signs and regulations as to what one can do or not; the last observed fact is that driving is only about going from an origin to a destination and just about driving. There is no lingering or wandering involved.

Walking and cycling on the other hand are very much dependent on physical strength and cognitive appeal of routes as well as social pleasance and safety. This latter aspect of safety is also completely different from the case of driving. One might go to some places in a car that would never dare to go without a car. We can assume that it would be safer to go on foot or by bike where there are more pedestrians and cyclists present. We argue that there is a need for an alternative framework for modelling walking and cycling accessibility that could address these issues.

§ 5.2 Research Background and Context

Urban spatial networks are mostly comprised of streets and some public open spaces. Topological structure of the spatial network can be generally modelled as either adjacency relations among junctions represented by point features (0D) or streets represented by line features (1D)[48]; these two categories can represent links between junctions or streets respectively. We refer to these categories[49] of spatial network representations as Junction-to-Junction and Street-to-Street graphs.

The first category is as old as Graph Theory itself[50] and is most common in transport modelling (Dios Ortuzar, J., & Willumsen, L. G., 2011), for it is convenient to measure metric distance on such models. This type of representation is also used in a number of spatial analysis models, namely Place Syntax (Ståhle A., Marcus, L. and Karlström, A., 2008), Urban Network Analysis (Sevtsuk, 2010), and Multiple Centrality Assessment (Porta S, Crucitti P, and Latora V, 2006a).

For taking into account the cognitive impedance of going from one street to another, the Street-to-Street adjacency models are more appropriate as they allow for attributing cognitive costs to links between streets. The most famous of this category of models is Space Syntax initiated by (Hillier, B., Hanson, J., 1984) and alternatives such as Named Streets (Jiang, B., & Claramunt C., 2004), Intersection Continuity Negotiation (Porta, S., Crucitti P., & Latora V., 2006), and Angular Analysis (Turner, 2007). Integrating physical and cognitive impedance in path finding has been researched before as reflected in (Hillier, B., & Ida, S., 2007), Place Syntax (ibid) and Multi-Modal Urban Network (Gil, 2014).

We have built upon the work of Turner (ibid.) and the Simplest Path of (Duckham, M., and Kulik, L., 2003) and developed an Easiest Path algorithm for finding the paths that are 'as flat, short, and straightforward as possible'. The optimal paths found by our algorithm allow for defining actual travel time or temporal distance and give rise to a number of accessibility measures.

What is particularly new in our approach is the way we model and aggregate costs, ensuring different costs are physically commensurate. Besides, taking account of topography in the same framework makes it distinctive from similar approaches. Therefore, we can redefine distances as 'actual' temporal distances experienced through easiest paths. Using these temporal distances, we provide a novel framework for accessibility measurements based on Fuzzy Sets theory (Zadeh, 1965). We then provide a Markov Chain model of walking and cycling flows using our graph model; and provide a fast algorithm for mathematically simulating Random Walks on such models. Covering our comprehensive notion of configuration, we model density distribution and land-use allocation processes in planning (as in defining spatial codes and regulations) based on configuration graphs.

§ 5.3 Definition of Urban Configuration

We define urban configuration graph as a labelled weighted graph that shows how roughly units of space are connected to one another. Weights of such a configuration graph would then represent the impedance or cost of travelling from one space (street space) to another. Important spatial attributes such as population density, built space density, or land-use mix can be spatially aggregated over the same spatial units that construct the nodes of such a configuration graph. The immediate advantage of such a representation is that it comes closest to the way we perceive our relative location to built environment: being in a place is recognized as being near to its neighbours. In our mental maps, we remember streets and urban public spaces as places that are located close to other places that we know.

FIGURE 70 the geographical map (left, 1909) the avant-garde topological map by Harry Beck [1902-1974] (right, 1933) of London Underground rail network, the so-called Tube, the oldest of the kind, and the first represented topologically, images from WikiCommons.

As can be seen in the figure above, the image on the right is definitely more intuitive than the one on the left. Ever since Harry Beck invented this type of representation most metro maps around the world have been represented in a similar way.

§ 5.3.1 Space versus Place: a Theoretical Clarification

3D Space can be modelled as Euclidean space using Cartesian coordinates in relation to a frame of reference. Alternatively, we can model space as topological space using topological definitions such as sets and their neighbourhoods (e.g. streets and their adjacent streets). This was nearly the initial approach put forward by Space Syntax 'axial' models (see the next section). Plainly, representing locations using abstract numeric coordinates has little to do with the way people perceive their location in space. The topological representation, however, is structurally closer to the way we humans make our mental maps of spaces. However, due to its high level of abstraction there will not be any direct indication of the actual physical place in the spatial model of Space Syntax. We argue that a place is recognizable by its relative location corresponding to other places in a geographical network and the impedance of its links to its neighbours, as well as its own particular spatial attributes. Clearly, such a definition depends on a definition of spatial units, but the point is that such a definition allows us to model the entirety of a city in such a way that is most relevant for urban design. This is to say, the relations amongst entities are more important than the entities taken in isolation (Hillier, 2007, p. 1). This is also a stance in mathematical modelling of cities, referred to as "network cities" (Portugali, 1999).

§ 5.3.2 Network Structure, Density, Diversity

Space Syntax theories (Hillier, B., Hanson, J., 1984) and (Hillier, 2007) relate the configuration of urban street network to urban functioning in terms of its effect on the distribution of densities and allocation of land uses such as retail and residence: "Land uses and building density follow movement in the grid, both adapting to and multiplying its effects. (Hillier, 2007, p. 127)" Hillier states that a spatially successful city is characterized by the "dense patterns of mixed use", which are mainly settled as a consequence of movement, which is itself "brought about by the grid configuration" (Hillier, 2007, p. 4). However, some important aspects of built environments are not addressed in Space Syntax models and analyses, namely: other modes of transport (e.g. train & metro), geographic attributes such as land-use and density[51], and the physical aspects of environment such as steepness of routes. For studying the effects of the structure of built environment on walking and cycling, we deem the city configuration as a structure that can be modelled with regards to its topology, geometry (regarding path continuity), topography, and the possible weight of locations because of their spatial attributes such as population density and land-use attractions.

Simply put, we can define urban configuration as a network structure that makes the connection between spatial units with such attributes as density-like of diversity-like indicators. Such networks intuitively show what is spatially accessible to a person who views the city from a location, i.e. from a human vantage point. For example, let us consider a map of a city that depicts population density per acre, hectare, or square kilometre and another one that shows the population density aggregated per streets. We can only go to places through streets, therefore our perception of city is formed by the way we can navigate it. In a street where there is little around us we, in a manner of speaking, feel being in the middle of nowhere, and in a street where there is a lot around us we feel like being in a crowded place. It is the density or diversity projected or perceived from within the streets that matters in our perception, preferences (say in buying or renting properties) and making such decisions as travel mode choice, and navigation.

§ 5.3.3 Reconstructing a Neighbourhood Mathematically

To illustrate our point, let us suppose that we want to model the dynamics of a neighbourhood in terms of dynamic flows of people, goods, and possibly information; say a static system and its dynamics. Modelling requires reduction and omission of many details, but the question is what kind of data needs to be in what model for what purpose? If we are to study mobility behaviour, transportation mode choice, or

TOC

accessibility, we definitely need to know the determinants of mode choices. In other words, we need to know what factors in the built environment might facilitate or hinder walking or cycling, and in particular choosing these modes of transportation over other options. Here we propose a few phenomenological assumptions.

We suggest that the ultimate choice on whether to walk or to cycle is for an individual to make. The exact such behaviour of individuals cannot be predicted. However, there are measureable factors that determine the feasibility or the relative utility of a journey on foot or by bike. We cannot predict every individual's action but we can measure these factors and understand the potentials of a built environment in terms of ease and feasibility of walking and cycling. We will be practically dealing with maps, which typically represent the geometry of environment together with a few spatial attributes such as population densities or land uses. There are common reasons as to why such information is typically most relevant and available. For the same reasons, we shall rely only on such information and seek to model the influence of the shape, configuration, and the topography of the built environment on walking and cycling accessibility.

Back to our topic, reconstructing a neighbourhood mathematically, we propose a thought experiment: Imagine that we have modelled a neighbourhood by representing its street network as an axial line map, typical for Space Syntax studies; the mathematical model underlying graph-theoretical measures of Space Syntax is merely an adjacency matrix that indicates what street is connected to which other streets. Now, let us assume that we want to re-present this model. We know that **a graph is an abstract entity that is not equivalent to its embedding or drawing**.

In other words, the graph representation of a spatial network as adjacency lists (or as a matrix) is much more abstract than the spatial network itself. We cannot reconstruct a spatial network from its adjacency matrix; meaning that the representation process is irreversible in that some vital information regarding the geometry of environment is lost in the process of reducing a street network to an adjacency graph[52]. This is in fact inevitable, for almost all network models. The more important question is whether an adjacency matrix could be enough for describing the 'actual travel impedances', e.g. for walking or cycling. If it is so, then the model could be good enough for studying the dynamics that are of interest. Once again, we should not forget that a model is not supposed to replicate the entirety of a complex system. In this case, such things as climatic conditions, greenness, pleasance, safety (from crime and dangerous traffic), or air quality of a path will definitely affect people's choices in their actual mobility behaviour; however, we do not have the verifiable mathematical means to incorporate such factors in our models. Therefore, we limit the scope of our model only to geometry, topography, topology, and geographic attributes of the environment.

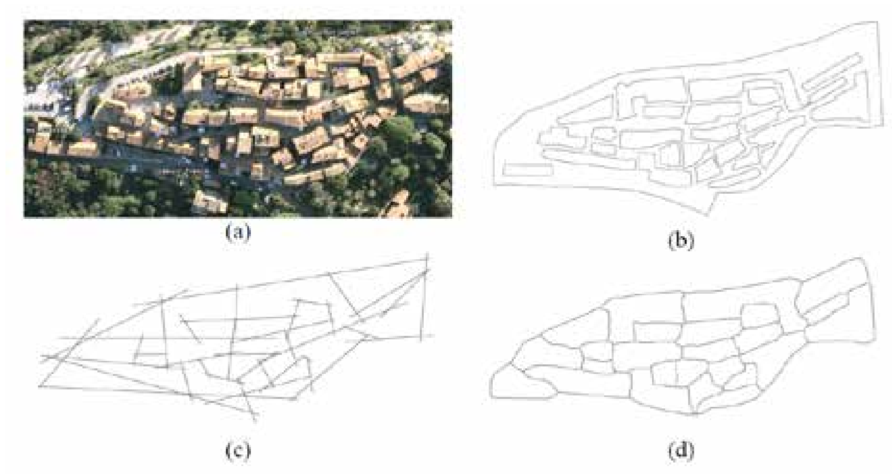


FIGURE 71 image from (Jiang, 2009) "(a) Gassin town image captured from Google Earth, (b) the open space, (c) the medial axes, and (d) the axial lines"

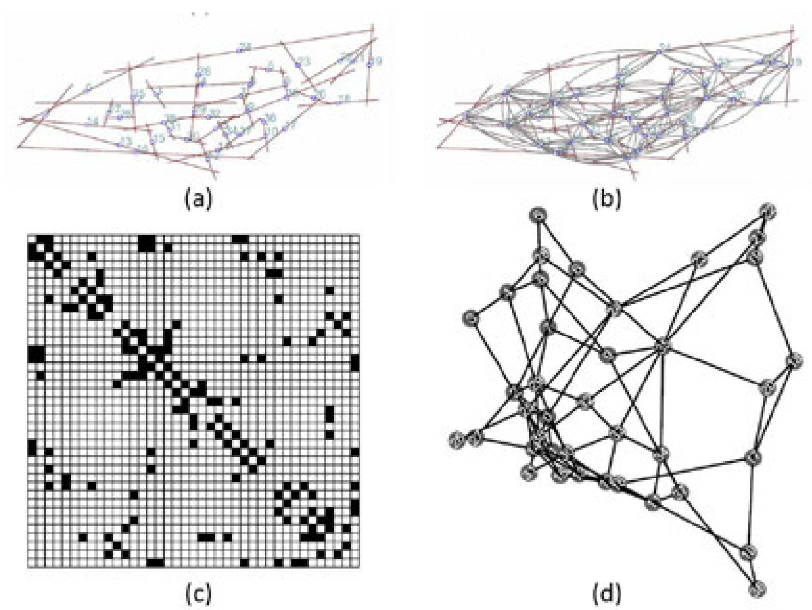


FIGURE 72 shows drawings of the axial line graph: (a) the initial axial lines of the Gassin village, (b) the arcs between axial lines showing their links, (c) a bitmap representation of the adjacency matrix associated with the axial line graph, and (d) a spectral graph drawing obtained from the Laplacian matrix associated with the same axial line graph.

§ 5.3.4 Matter of Scale in Urban Analysis: Global vs. Local

One of the subtlest issues in network studies is the matter of scale. In particular, speaking of such things as centrality or accessibility in absolute sense could be problematic prior to clarifying the scale of a network analysis. It can be observed that apart from special cases of islands, most built environments are ultimately connected to many others; so, how can we choose the right boundaries for a network model prior to a network analysis, e.g. on centrality or accessibility of locations? In other words, how sensitive are our measurements to the scale of spatial network underlying our analyses? A proper/generic answer to such questions requires extensive empirical research that falls beyond the scope of this research; however, we adopt certain strategies to avoid such issues. In particular, using our Fuzzy Logics framework, we map the accessibility of all locations far away from a place as absolutely far and thus disregard them consistently from our centrality or accessibility calculations. Following this strategy, we only need to set a properly sized buffer area around the site under study to avoid such sensitivities. This approach has an effect similar to the approach of Hillier in formulating Local Integration in Space Syntax, in which the nodes taken into calculation of integration are chosen within a step-depth radius of the node in question.

§ 5.3.5 A Chicken and Egg Problem

Do structural properties of a space determine its usage or spatial function? Why do some spaces become attractions in networks -because of their network-structural properties or because of their function? In other words, are 'land-use allocation patterns' shaped according to network structure? When a politician or a developer decides to build something somewhere, they do not necessarily perform a network analysis to see if the proposed building would 'function' in the proposed context. Obviously, many new architectural developments fail miserably in addressing their claimed purpose. Many examples can be mentioned where there is a clear mismatch between the network-related potential and space usage, from the so-called community centres that instead of attracting families attract criminals and youth gangs to shopping centres that never work as supposed to. We argue that the question is in fact irrelevant and that there is no such paradox. In case of organically grown settlements where top-down planning and design has not systematically happened, we can perhaps find matching patterns between network structure and the distribution of densities and diversities. However, in the case of contemporary built environments, this match might be less strong and even completely lost. The idea is not necessarily to make a perfect match between all spatial patterns, but to find a way to see if a spatial configuration has the “affordance”[(Gibson, 1977) & (Gibson, 1979)] to facilitate the desired functional patterns. Thinking of affordance as the ‘probability’ that most people would behave in a certain way in the environment, we can think of a configurative design process as a way to increase the probability of certain spatial behaviours knowingly.

§ 5.4 Urban Spatial Network Modelled as a Configuration Graph

Constructing a street network that is topologically clean and valid is not trivial when dealing with real datasets such as OpenStreetMap[a]. In fact, the subject of constructing topological data models of networks is rigorously studied specially for automating such procedures, e.g. see the methods for constructing network topological models for traffic simulations as in (Nielsen, OA, Israelsen, T & Nielsen, ER, 1997). We address this issue only out of necessity; thus, it suffices to mention a number of techniques we make available in our implementation of our methodology (see the next chapter).

In our work, we have used disks (topological neighbourhoods) to establish distinctness of points and incidence between points and lines. We adjust the precision of the road centrelines (polylines) and their vertices through a process of topological voxelization (Laine, 2013), remove pseudo nodes (nodes that do not indicate a street junction); split the street polylines at junctions; and insert nodes at junctions. By removing duplicate points, we form a list of vertices and by finding incident lines to these vertices; we form a list of topological edges between these vertices[53]. The lines can be drawn in both directions to allow for construction of a Point-to-Point directed network. The edges of this network become the nodes of the Line-to-Line graph representation that we use for finding Easiest Paths.

Practically, one has to have many tools for making a valid network model from a real-world dataset due to many often-present problems that make such datasets topologically invalid or not immediately usable. Automatic validation of such datasets for different purposes requires corrections that cannot be always fully automated. Best approaches could notify the user of the whereabouts and existence of data problems such as isolated streets, dangling edges and alike; but diagnosis is one thing and treatment is quite another. It is, in most cases, not possible or meaningless to automate such corrections before knowing the exact situation as to which the model is built.

a https://www.openstreetmap.org

Therefore, human intervention is often required at some point in the insertion and validation of a network model. This is in fact one of the reasons that an urban network analysis toolbox is most operational in a GIS or CAD environment where doing manual edits on points and lines is easy and manageable and at the same time proposing design alternatives is most convenient. Moreover, handling vital 3D information in a CAD environment is much easier than most GIS packages.

Prior to technical challenges, we need to reflect upon how and why we are reducing an actual spatial network to a geometric network of curves or lines. The most important issue in this regard is the fact that walking and even cycling movement trajectories are not exactly bound to streets as we can observe in the case of vehicular movement. While acknowledging the fact that pedestrians and cyclists have more freedom to deviate from street lines, we should face the limitations of our chosen models in terms of space-time complexity of computation. A perfectionist approach to modelling pedestrian movement within space would require implementing a raster representation of space or a fine resolution triangulation in order to account for the mentioned freedom in choosing paths. However, considering our ultimate goal, i.e. acquiring insight on the relation between built environment configuration and mobility, it would be in fact better to work with a more simplified abstraction of space. Let us clarify this stance with an example. Suppose we are to study the pedestrian movement patterns in a city through simulation and comparing the simulation results with ground truth data acquired using GPS devices. We will have to do some cleaning on the GPS trajectories acquired and somehow snap them to discrete units of space in a spatial network model. Note that computing is essentially about dealing with discretized models. Now, if we use a fine resolution raster as the spatial model and somehow manage to snap the trajectories to this model then we can seek for some associations between our simulation results and the ground truth data. If we want to say something about particular streets, then we have to aggregate our analysis results to gain some information about the streets. This would require averaging of the data acquired for many pixels or voxels to be able to say something about an underlying surface.

In light of this example, we can see that it would be in fact better to start with a coarse resolution model from the beginning, so that our spatial units would come closer to what we would like to study eventually, i.e. the streets.

The next important question regards ways of representing a street and representing the adjacencies between streets. Consider the fact that a model is necessarily a reduced replica of something in reality. In this case, if we are to represent a street geometrically, we can represent it as a 1D polyline 2D surface (TIN), or even a 3D poly-surface. Regarding our ultimate goal of representing, analysing and modelling movement trajectories, we can justify the choice of polylines or other 1D data models for representing streets. Here comes the next question: how do we represent a street network using line segments?

The approach of Space Syntax is using the so-called Axial Lines. The axial lines in their first definition [in (Hillier, B., Hanson, J., 1984)] were supposed to represent their underlying convex spaces (Batty, M, Rana, S, 2004). This inexact definition has later transformed into formulations such as 'a minimum set of lines that are supposed to represent the longest lines of sight[a] in an environment' (ibid). Space Syntax models have been criticized primarily because of the inherent difficulty of arriving at such a representation clearly and in a rigorous and repeatable manner (ibid, also in (Ratti, 2004)). There have been a number of attempts to suggest alternative algorithmic definitions of axial lines and for producing an 'axial line map' representation of spatial networks, namely (Batty, M, Rana, S, 2004), (Jiang, 2009), (Jiang, Bin, and Xintao Liu., 2010). The importance of Axial Lines or Convex Spaces is primarily pertained to human perception of space, in that these representations can potentially help us reconstruct space in the way it is potentially perceived by human beings.

To clarify this importance, let us see what would be problematic in using other conventionally available representations such as street centreline models used for transport modelling. In such models, it is typical to have a few lines representing the two sides of a street and possibly such things as bus or tramlines. However, if we are to research the relation between spatial perception and spatial behaviour of pedestrians and cyclists then such lines are more of a problem than a solution. Thus, a purer representation of street space would then be more desirable. This line of thinking draws our attention to methods that can represent a 'topological skeleton' of an open space, which can be modelled as a polygon with holes. A number of different approaches can be followed for this purpose, namely convex decomposition, Medial Axis Transform [as in (Miranda, P, and Koch, D, 2013)], approximated Segment-Voronoi [as in (Sileryte, 2015), see figure below], and Straight Skeleton (Aichholzer, O., Aurenhammer, F., Alberts, D., & Gärtner, B., n.d.).

a The term line of sight refers to the concept of visibility for potential observers.

FIGURE 73 shows a segment-Voronoi diagram used to draw a street centreline map (Sileryte, 2015).

Giving a proper account of spatial network representations falls outside of the scope of this dissertation. However, we claim that the potentially most suitable approach for our purpose would be using Straight Skeleton method [(Aichholzer, O., Aurenhammer, F., Alberts, D., & Gärtner, B., n.d.) & (Felkel, Petr, and Stepan Obdrzalek, n.d.)] presented on an example urban environment. Later in the next chapter we will discuss our approach for validating the topology of a given street centreline network, which could be imported from OSM[a] by users of our application CONFIGURBANIST (that implements our configuration analysis methodology).

Suppose we have a line-network representation of an environment. We can represent this line-network as a graph in different ways. In street network studies, there have been two main categories of street network representations, namely Junction-to-Junction and Street-to-Street graph representations –loosely speaking[54]. In the first category, street

a https://www.openstreetmap.org

junctions are considered as nodes and streets as links, whereas in the second category, streets are considered as nodes and junctions as links between them. Geometrically, the former representation represents adjacencies between points (0-Dimensional features) and the latter represents adjacencies between lines (1-Dimensional features like axial line, centreline, or a curve representing a street). This is of course an over-simplified way of describing these two categories; we refer the reader to a more extensive review of network representations for syntactic studies to (Batty, 2004). In a Point-to-Point representation, which is as old as graph theory itself[55], it is quite straightforward to measure physical distance between locations and thus it is the de facto standard of transportation studies (Dios Ortuzar, J., & Willumsen, L. G., 2011). On the other hand, Space Syntax in particular and some other approaches such as Intersection Continuity Negotiation (Porta, S., Crucitti P., & Latora V., 2006) and Named Streets (Jiang, B., & Claramunt C., 2004) model street network as a Street-to-Street adjacency graph[56]. The immediate advantage of this approach to spatial network representation is that the notion of location is associated with streets and therefore the whole representation comes closer to the way people perceive their location in cities. In addition, cognitive aspects of way finding can be modelled more easily on Street-to-Street representations because usually there is cognitive impedance in going from one street to another, which affects way finding as in crossing a junction (the feeling of getting away from an origin) towards a destination. It sounds reasonable to attribute the relative success of Space Syntax in indicating pedestrian mobility to this fact; as walking requires intuitive way finding. It is important to note that Turner (Turner, 2007) also builds his remarkable angular analysis method upon a Line-to-Line adjacency representation of street network. In spite of the differences between an axial line representation of a street by a centreline, (or other variations such as the one introduced by Jiang & Liu (Jiang B, Liu C, 2009) these representations are all classifiable as Street-to-Street adjacency graph representations. The drawback of most of such representations is that the physical distance between locations is not taken into account.

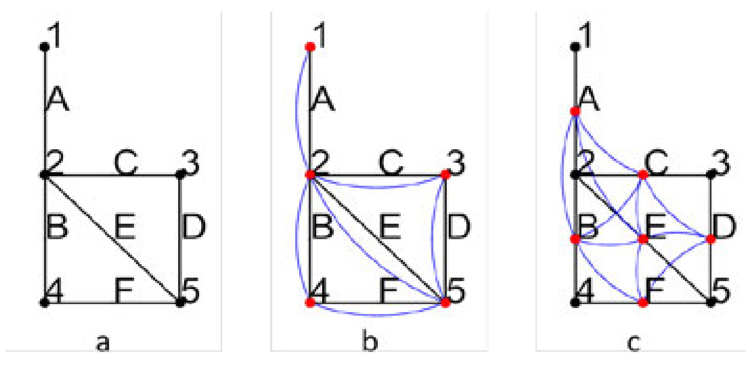


FIGURE 74 A hypothetical street network (a), a Junction-to-Junction adjacency graph (b) versus a Street-to-Street adjacency graph (c), both 'undirected', after Batty (Batty, 2004): red dots represent graph nodes, and blue arcs represent graph links.

TOC

We started our research with the assumption that in way finding for pedestrians and bicyclists and constructing a simulation model of walking/cycling flows, we need to take into account both the physical aspects of mobility and the cognitive aspects of way finding. Combining physical and cognitive aspects of way finding into a single model is previously researched by a few scholars as in the Multi-Modal Urban Network (Gil, 2014), which is based on an undirected Street-to-Street graph representation, or Place Syntax (Ståhle A., Marcus, L. and Karlström, A., 2008), which combines topological distance with metric distance. In addition, Hillier and Ida (Hillier, B., & Ida, S., 2007) have compared different distance measures (metric, angular and topological). Building upon the work of Turner (ibid.) and (Duckham, M., and Kulik, L., 2003) we have constructed a new Street-to-Street 'directed' graph representation that incorporates both physical and cognitive distance into its graph definition. Based on this representation we form a new geodesic (optimal path) that we call the 'Easiest Path' for walking or cycling. The following image briefly introduces these models.

Note that it is necessary to construct a dual graph for modelling impedances pertained to traversing one line feature to another line feature. This is because we can solve an optimal path problem when formulated as a minimum weight path, i.e. a path consisted of links whose sum of weights or costs is minimal. We shall present a formal definition of this formulation later in this chapter.

Given a street network (as a set of lines and points at their junctions), we need to define a graph representation on which we can run a graph geodesic algorithm to find the best paths. We call every segment made up of streets an edge e and each endpoint of these edges as a vertex v; therefore, we denote this segment network G as an ordered set of vertices and edges:

$$\mathrm{G}(\mathrm{V},\mathrm{E}): \mathrm{v} \in \mathrm{V} \;\&\; \mathrm{e} \in \mathrm{E} \tag{30}$$

As mentioned above, we have chosen to use a dual graph representation, which is by the way tightly related to the primal graph representation and the definition of an especial adjacency matrix that describes how each vertex is connected to some edges in G. We follow the approach proposed by Michael Batty (Batty, 2004) in forming this graph. Considering a graph as an abstract mathematical description of how some nodes are linked to one another, we are free to choose either set of V or E in G as the set of nodes for this graph and the other as the set of links. We call this graph the Configuration Graph and denote it by $\Gamma(\mathrm{N},\mathrm{L})$ as an ordered pair of nodes N and links L.

$$\text{primal adjacency graph: } \Gamma_{\mathrm{p}}(\mathrm{N},\mathrm{L}): \begin{cases} \mathrm{N} \stackrel{\text{def}}{=} \mathrm{V} \\ \mathrm{L} \stackrel{\text{def}}{=} \mathrm{E} \end{cases} \tag{31}$$

$$\text{dual adjacency graph: } \Gamma_{\mathrm{d}}(\mathrm{N},\mathrm{L}): \begin{cases} \mathrm{N} \stackrel{\text{def}}{=} \mathrm{E} \\ \mathrm{L} \stackrel{\text{def}}{=} \mathrm{V} \end{cases} \tag{32}$$

Considering the rectangular (not necessarily square) incidence matrix of $\boldsymbol{A}_{VE}$ as the descriptor of incidences between vertices and edges in G we can see that:

$$\mathbf{A}_{\mathrm{VE}} = [\mathrm{a}_{\mathrm{i,j}}^{\mathrm{ve}}]_{|\mathrm{V}|\times|\mathrm{E}|} = \begin{cases} 1 \text{ if } \mathrm{V_i} \sim \mathrm{E_j} \\ 0 \text{ otherwise} \end{cases} \quad (33)$$

$$\mathbf{A}_{\mathrm{EV}} = [\mathrm{a}_{\mathrm{i,j}}^{\mathrm{ev}}]_{|\mathrm{E}|\times|\mathrm{V}|} = \begin{cases} 1 \text{ if } \mathrm{E_i} \sim \mathrm{V_j} \\ 0 \text{ otherwise} \end{cases} \quad (34)$$

It can be seen that:

$$\mathbf{A}_{\mathrm{VE}} = \mathbf{A}_{\mathrm{EV}}{}^{\mathrm{T}} \quad (35)$$

In addition, we can deduce intermediate adjacencies (coincidences) between vertices and other vertices in A_{VV} and intermediate adjacencies between edges and other edges in A_{EE}. After (Batty, 2004) we write the interlocking equations below:

$$\mathbf{A}_{\mathrm{VV}} = [\mathrm{a}_{\mathrm{i,j}}^{\mathrm{vv}}]_{|\mathrm{V}|\times|\mathrm{V}|} = \begin{cases} 1, & \text{if } \mathrm{i} \neq \mathrm{j} \ \& \ \mathrm{V_i} \sim \mathrm{V_j} \\ \mathrm{Deg}(\mathrm{v}), & \text{if } \mathrm{i} = \mathrm{j} \end{cases} \quad (36)$$

$$\mathbf{A}_{\mathrm{EE}} = [\mathrm{a}_{\mathrm{i,j}}^{\mathrm{ee}}]_{|\mathrm{E}|\times|\mathrm{E}|} = \begin{cases} 1, & \text{if } \mathrm{i} \neq \mathrm{j} \ \& \ \mathrm{E_i} \sim \mathrm{E_j} \\ \mathrm{Deg}(\mathrm{e}), & \text{if } \mathrm{i} = \mathrm{j} \end{cases} \quad (37)$$

It can be shown that:

$$\mathbf{A}_{\mathrm{VV}} = \mathbf{A}_{\mathrm{VE}}\mathbf{A}_{\mathrm{EV}} \quad (38)$$

$$\mathbf{A}_{\mathrm{EE}} = \mathbf{A}_{\mathrm{EV}}\mathbf{A}_{\mathrm{VE}} \quad (39)$$

Where $Deg(v)$ denotes the number of vertices intermediately (through a single intermediary edge) adjacent to a vertex v and $Deg(e)$ denotes the number of edges intermediately (through a single intermediary vertex) adjacent to an edge e.

We denote $\boldsymbol{A}_p$ as the adjacency matrix corresponding to the primal graph $\Gamma_p(\mathrm{N},\mathrm{L})$ and $\boldsymbol{A}_d$ as the adjacency matrix corresponding to the dual graph $\Gamma_d(\mathrm{N},\mathrm{L})$; in addition, we consider diagonal matrices $\boldsymbol{D}_p$ and $\boldsymbol{D}_d$, whose diagonal entries are respectively equal to degrees of vertices and edges in G.

$$\mathbf{A}_{\mathrm{VV}} = \mathbf{A}_{\mathrm{p}} + \mathbf{D}_{\mathrm{p}} | \mathbf{D}_{\mathrm{p}} := \text{a diagonal matrix containing row sums of } \mathbf{A}_{\mathrm{VE}} \quad (40)$$

$$\mathbf{A}_{\mathrm{EE}} = \mathbf{A}_{\mathrm{d}} + \mathbf{D}_{\mathrm{d}} | \mathbf{D}_{\mathrm{d}} := \text{a diagonal matrix containing row sums of } \mathbf{A}_{\mathrm{EV}} \quad (41)$$

It is of course quite straightforward to obtain $\boldsymbol{A}_p$ and $\boldsymbol{A}_d$ from $\boldsymbol{A}_{VV}$ and $\boldsymbol{A}_{EE}$ computationally. If we generalize the configuration graph $\Gamma(\mathrm{N},\mathrm{L})$ to a weighted graph whose links have 'impedance weights'[57] $\zeta \in \mathrm{Z}$ (i.e. each link having a cost or impedance of traversal) then we can apply a graph search algorithm on $\Gamma(\mathrm{N},\mathrm{L},\mathrm{Z})$ to find optimum (minimum cost/impedance) paths between arbitrary origins and destinations. It is clear that the cardinality of L and Z are the same and therefore we use the same indices for numbering links and their weights/costs.

A general definition of an adjacency matrix would indicate the number of paths of length one in between two nodes of the network. In line with this definition, we can forget about topological vertices & edges; and think of two types of elements related to their counterparts through incidence (relation between an element of one type to an element of the other type) and adjacency (relation of an element of one type to another element of the same type). Graphs made as such do not necessarily have a planar embedding and their adjacency matrices might include entries larger than one in non-diagonal positions.

§ 5.4.1 Primal Undirected Graph, Undirected Network

This type of spatial network model was our first approach to the problem of measuring walking and/or cycling accessibility. Because of the inherent shortcoming of this approach in representing the traversal impedance between streets (as 1D features), we discontinued model and implemented two alternative dual graph representations of urban configurations as follow.

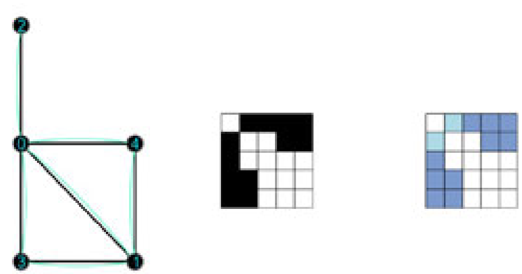


FIGURE 75 an undirected network of lines and a primal graph describing how the junctions of these lines are adjacent to each other. To the right, the adjacency matrix and the weighted adjacency matrix pertained to this graph are shown as bitmaps. The bit maps are laid out from top left corner, i.e. the top left pixel is depicting the (i,j) entry (0,0), and its right neighbour the entry (0,1) and so forth.

§ 5.4.2 Dual Directed Graph, Doubly-Directed Network

Our first implementation of the Easiest Paths algorithm is based on a doubly directed dual graph model that considers each street as traversable in both directions. This construct allows for modelling different impedances for uphill and downhill directions on the same street (also one-way streets, if necessary). Considering cases in which walking and cycling trips are not necessarily commute-like, e.g. walking to a metro station, riding a bike from an urban bike-sharing system downhill, then this approach can be more precise. This precision however comes at a high cost of computation because the size of the adjacency matrix associated with such an adjacency graph is double the size of dual graph associated with an undirected network. This means that for searching this graph, e.g. for all geodesics, even with the most efficient search algorithms such as Floyd Warshall, which has the complexity of $O(n^3)$ the running time would be eight times longer, which could be prohibitively long for large networks. Avoiding senseless detours at Y-shaped junctions (a.k.a. T-Junctions, shown in Figure 76 pointed to by (Turner, A., and N. Dalton, 2005) and (Duckham, M., and Kulik, L., 2003).) is one of the reasons to implement a directed graph upon a doubly directed edge network. The other reason is to allow for different impedances for uphill and downhill sides of a road segment.

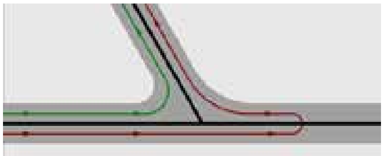


FIGURE 76 is reproduced after Turner and Dalton (Turner, A., and N. Dalton, 2005). It shows a detour at a Y-shaped junction. If we do not have a way to differentiate between 0° & 180°, i.e. running a search algorithm on an undirected line network, we will encounter this problem. By implementing our two graph models this problem will not occur.

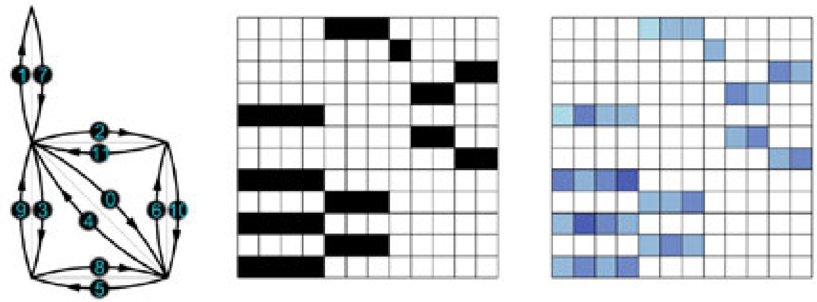


FIGURE 77 a doubly directed network of lines representing streets within which one can move in both directions. The bitmaps at the right respectively shown the adjacency matrix and the weighted adjacency matrix associated with the dual graph constructed from this network. Graph links are not shown in this image for the sake of simplicity. See the next figure for the directed links.

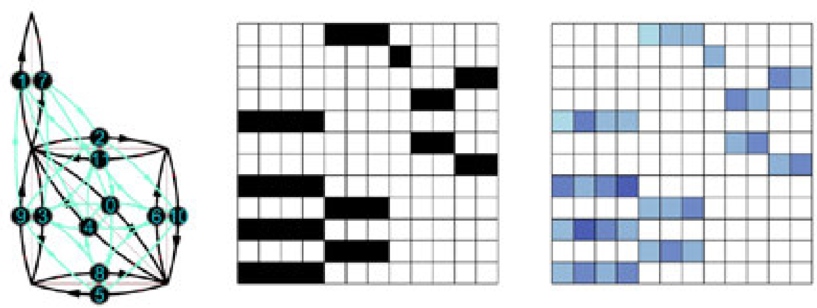


FIGURE 78 shows the links of the previous directed graph constructed on a doubly directed network of lines. Note that the adjacency matrices associated with this graph are not symmetric as to the directedness of the underlying graph.

§ 5.4.3 Dual Directed Graph, Undirected Network

Using the doubly directed network as discussed is one way of solving the problem of meaningless detours; but it comes at a very high cost of doubling the size of network and complicating further centrality analyses. Therefore, for improving the theoretical time-complexity of the overall search for all optimal paths, we decided to construct a method for computing Easiest Paths on a dual graph constructed upon an undirected network. Although it might sound an easier modelling task, in practice there is a bigger challenge in making such a graph useful: how can we consistently measure azimuth angles between streets in order to differentiate between meaningless U-turns (180°) and a continuation without direction change (0°)? Failing to address this issue will result in irrational trajectories involving U-turns, as In light of this necessity, our method for computing shortest angular paths without doubling the size of network (quadrupling the size of graph) is a breakthrough that makes finding Easiest Paths comparable in terms of time-complexity to ordinary shortest paths. Note that going from a street to another street would cause the same angular confusion in both directions. What could make the graph directed is then then the downhill or uphill slopes in opposite directions.

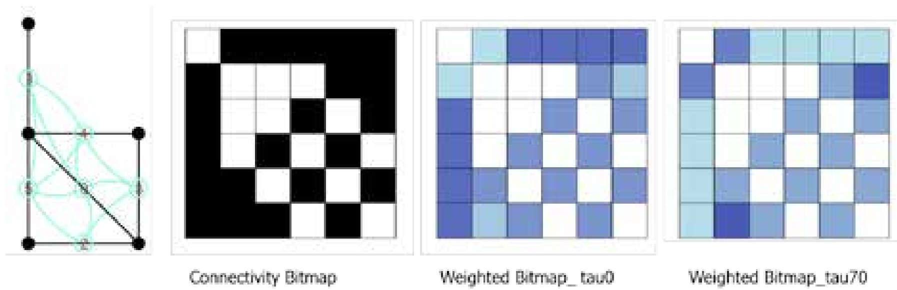


FIGURE 79 shows from left to right an undirected dual graph on an undirected line network, which could be directed itself, its adjacency matrix and its weighted adjacency matrices with different tau values. By implementing an advanced angle-computer that differentiates between acute and obtuse angles, we assign weights correctly, as apparent in the rightmost image: the connection between 1 and 5 is straightforward and thus strong. Note that our model is generally directed.

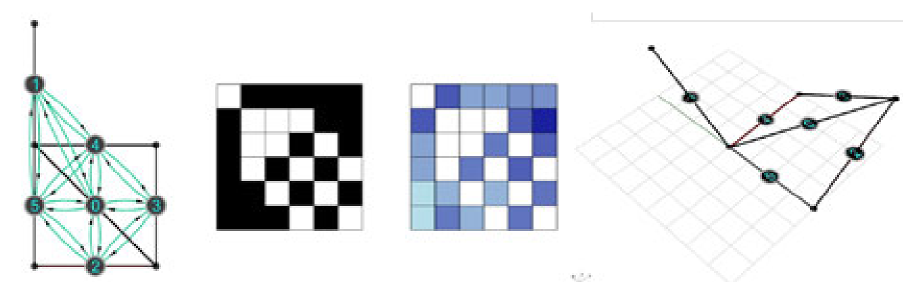


FIGURE 80 shows a dual directed graph that is the final configuration graph constructed and used in our implementation of the Easiest Path algorithm as well as the underlying spatial configuration for the random walk models. This graph is directed (has asymmetric impedances) because of uphill downhill differences as shown at the right hand side image.

§ 5.4.4 A Unifying Framework (Architectural, Urban, Spatial)

What is particularly interesting about the last graph representation is that it is in fact the same as the representation used in modelling spatial networks within buildings. As explained in Chapter 3, the dualities between spatial features (different in 2D from

3D) are used to model a spatial network. Suppose, for example, the navigable space within a building is represented as a cell division whose cells represent the units of space (e.g. as convex cells); these cells as 2D faces have their duals as 0D vertices and the edges that connect these faces have their duals as crossing edges that connect the dual vertices. Considering the fact that in our urban configuration model we take streets (represented as 1D features) as units of space and their adjacencies through the same duality principle we can observe that the principle and the approaches used in both cases are the same. The only difference is the inherent topological dimensionality of the street network representation that is 1D. In light of these facts, we can claim that we can treat architectural and urban spatial configurations using our unified framework that is based on Poincare duality theorem. The framework is of course supported by the dual directed graph representation explained above.

This is to say it will be possible to treat urban and architectural configurations using one set of methods and tools. This is the motivation behind developing **configraphix.dll** as a shared library for spatial configurational computations.

§ 5.5 Way Finding and Geodesics in Urban Environments

There are several criteria affecting one's choice of walking and cycling routes, which might have different levels of importance. In order to have an all-inclusive model we need to study each factor as cost attributes on a street-to-street adjacency graph representation. Aggregation of costs of different physical natures is an issue that must be treated properly by taking into account the physical dimensionality and meaning of cost attributes. This issue is referred to as commensurability in physics. It might seem very practical to model such things as beauty of roads and their popularity and attributed some numeric measures as to such qualities but these types of measurements cannot be compared or aggregated with one another, let alone their meaning in themselves. We have to accept that a mathematical model for finding a path of minimum costs can only serve to find a path that ensures a minimum sum of costs that can be measured as physically meaningful quantities. We prefer to work such quantities because their interpretation will be straightforward. Turner and Dalton (Turner, A., and N. Dalton, 2005) have discussed in detail the necessity of considering directionality by referring to evidences and examples from transportation geographic information research. We, therefore, shorten the discussion and refer the interested reader to their work for more information.

§ 5.5.1 Walking and Cycling Impedance in Built Environment

We assume that in walking and cycling it is important to consider both the physical difficulty of travelling, relying on human power, as well as the cognitive difficulty of way finding or navigation. In reality, other factors also affect route choice of pedestrians and cyclists, e.g. social safety, climatic pleasance, pollution, traffic safety, or scenery. We have not yet succeeded in integrating them mathematically within our model mathematically. It is important to note that we can search for 'minimum cost' paths, therefore if we are to include one aspect in our route choice model we should be able to define it as a measure of impedance commensurate to others (i.e. of the same physical dimension as others)[58].

An important issue here regards the ways in which impedance values of different natures can be aggregated in order to choose a path that is best in all respects. The problematic aspect of such aggregations is the matter of commensurability of costs, which is about avoiding comparisons (including additions and subtractions) of quantities of different physical dimensions (e.g. comparing apples and oranges). This issue seems to have been neglected in the literature in combining different costs for path finding. An example is the improper use of a weighted sum model in dealing with costs of different nature in (Nagar, Atulya, and Hissam Tawfik, 2007). To illustrate the problem, let us imagine that the two types of costs are monetary and metric: can we add 2$ and 2 metres? Suppose we do so, then what would be the meaning of this sum? What would happen if we compare it with another such sum? Adding and subtracting quantities of different physical dimensions is senseless. Of course, the same problem holds for averaging them using a Weighted Sum Model.

Our first alternative was to use a Weighted Product Model to allow for integrating costs of different nature, because there is no restriction in multiplication of quantities with different dimensions. However, as it turns out, the cognitive impedance when computed based on angular change of direction can be zero or close to zero. Therefore, a Weighted Product Model cannot be helpful, for it overlooks other costs when the directional change is negligible. Alternatively, we have chosen to make the costs commensurate. We do so by measuring all costs in terms of travel time. It is clear that the actual travel time of a pedestrian or cyclist is not only dependant on their physical strength but also on their cognitive ability for navigation. If the person in question is not familiar with the environment, they are to lose some time in finding their way towards a destination. We can also observe that the actual trajectories of people walking and cycling in cities is not the same as shortest paths, for people seem to prefer intuitive paths, i.e. paths that require less cognitive effort in navigation. Speaking of commensurate impedance measures, we introduced a parameter to consider a so-called 'confusion time' proportionate to a Fuzzy model of cognitive difficulty of navigation at junctions.

§ 5.5.1.1 Physical Impedance: how long and how steep

When walking or cycling, people depend heavily on their physical strength, this limitation determines the effective speed of walking or cycling that is easy to maintain and thus affects the temporal distance to certain destinations and eventually willingness to take or not to take certain routes. We can calculate such mobility speeds and relate them to the slope of road segments by taking into account the physical power that a normal person can easily maintain. Obviously, this power would be a parameter that can be adapted to represent the conditions of those with less strength or those riding on power-assisted bikes such as e-bikes. Without loss of generality, we assume that an average person can maintain a power of around 100 Watts easily for about an hour or so. We have modelled walking speed as a function of slope by Waldo Tobler (Tobler, 1993) as shown in Figure 82. Inspired by the model of Tobler (ibid), and the illuminating blog article of Rhett Allain (Allain, 2013) on the physics of cycling, we have formulated a model of cycling speed as a function of slope angle (only uphill slopes for the time being).

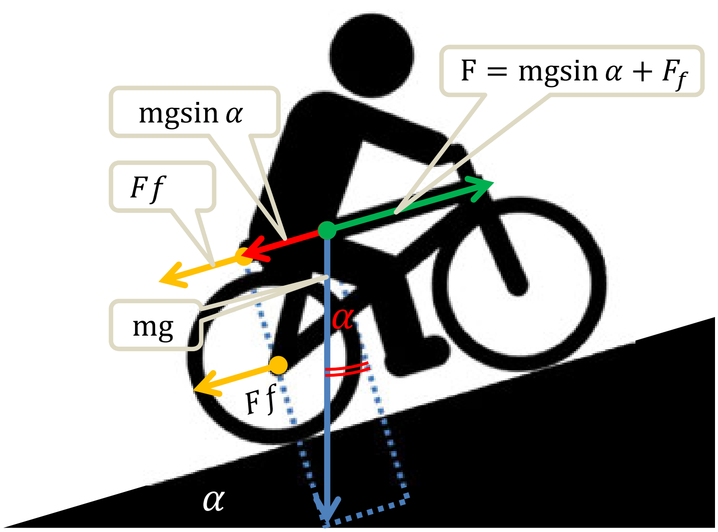


FIGURE 81 shows how forces act upon a bike, m=bike's mass+ body mass, F_f = rolling friction force, g=gravity acceleration, and α=slope angel, image produced after (Allain, 2013).
Image reproduced after the image found on this URL: http://www.lloydswellbeingcentre.co.uk/clives-cycling-blog-18

Considering that, a cyclist can sustain a power P, which corresponds to the gradient of energy expenditure over time from the cyclist to exert a force of $F = mg\sin\alpha + F_f$ at the speed of CV standing for cycling velocity. Formally, the power required for moving an object with force F and velocity V can be measured as the dot product of force and velocity vectors:

$$\mathrm{P} = \mathbf{F}.\mathbf{V} = |\mathbf{F}||\mathbf{V}|\cos\varphi \xrightarrow{\varphi=0\ ,\mathrm{V} \stackrel{\mathrm{def}}{=} \mathrm{CV},\text{and } \mathrm{F}=|\mathbf{F}|} \mathrm{CV} = \frac{\mathrm{P}}{\mathrm{F}} \qquad (42)$$

The force F is to compensate for the friction force and the weight of the bike and the rider projected along the path of movement.

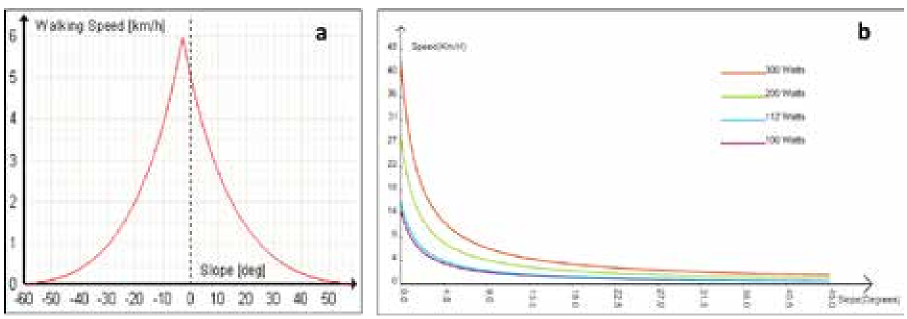


FIGURE 82 Picture (a) shows a graph of Tobler hiking function. Note that at the slope of 0 the walking speed is 5 Km/h. Also, note that on downhill slopes humans do not walk much faster. In fact, a little bit of downhill slope boosts walking speed but too much of downhill slope slows down walking. Picture (b) shows plots of our model of cycling speed as a function of slope at the constant power of 112 Watts, which is approximately 0.15 HP, a power that an average human can easily sustain for about two hours.

The ultimate aim is to obtain a model of temporal cost of traversing a segment in terms of its slope angle. We consider such costs as impedances physically hindering mobility and denote them as $\zeta^{W}_{(i,j)}$ and $\zeta^{C}_{(i,j)}$ as for walking impedance and cycling impedance of the (i,j) link (between i^{th} and j^{th} nodes, which are street edges)[a]. In these equations, δ represents the displacement along the (i,j) link and α denotes the slope angle of the link in radians, m the mass of an average person typically assumed to be 75 Kilograms and g for gravitational acceleration equal to $9.81\ \mathrm{m/s^2}$, and $\mathrm{F_f}$ denoting a nominal force of friction that is to be counteracted by the bicyclist[59].

a We remove the subscript (i,j) for the sake of clarity of our notations

$$\delta = \frac{|e_i| + |e_j|}{2} \text{ and } \alpha = \text{Elevation} \angle(e_i, e_j) \cong \sin^{-1}\left(\left|\frac{\Delta Z}{\delta}\right|\right), \Delta Z = \overline{e_j.Z}\text{-}\overline{e_i.Z} \tag{43}$$

$$\zeta^{W}_{(i,j)} := f(\alpha, \delta) = \frac{\delta}{V^{W}(\alpha)} = \frac{3.6\delta}{6e^{-3.5|\tan\alpha+0.05|}} = \frac{3.6\delta e^{3.5|\tan\alpha+0.05|}}{6} \tag{44}$$

$$\zeta^{C}_{(i,j)} := g(\alpha, \delta) = \frac{\delta}{V^{C}(\alpha)} = \frac{\delta(mg\sin\alpha + F_f)}{P} = \frac{\delta(85\times 9.81\times \sin\alpha + 25)}{112} \tag{45}$$

The two impedances ζ^W and ζ^C are computed in terms of [average] seconds it takes one to traverse a link. We need to note that we have not yet considered the higher speed of pedestrians or bicyclists on downhill roads.

§ 5.5.1.2 Cognitive Impedance: how difficult to navigate

We consider change of direction or turning at each junction, as a cognitive kind of impedance for a pedestrian or bicyclist traversing that link. We denote angular impedance of $(i,j)^{th}$ link as $\zeta^{A}_{(i,j)}$, which is then simplified as ζ^{A}. In order to compute this, we need to find the angles between nodes (street segments); then we need to attribute impedance values corresponding to these angles consistently. We find the angles as shown in Figure 84. In order to make the dimension of ζ^A commensurate[60] with those of ζ^W or ζ^C we need to introduce a 'temporal confusion' coefficient in terms of the seconds it would take a person to take a decision as to which street incident to the junction has to be followed next. By adjusting this parameter, denoted as τ, we can distinguish between those who know the neighbourhood well from newcomers and tourists. It is important to note that in our method we have chosen to disregard angular impedance for the junctions of degree 2 (at junctions in between only two street segments) because, we believe that they do not cause any significant cognitive impedance compared to junctions between multiple streets where making a choice requires a bit of thinking. In other words, these simple 2-street junctions cause no confusion when it comes to directionality. The angle θ is the planar angle between i^{th} and j^{th} street.

$$\theta_{(i,j)} = \begin{cases} \theta, \text{if accute} \\ \pi\text{-}\theta, \text{if obtuse} \end{cases} | \theta = \cos^{-1}\left(\frac{\boldsymbol{u}_i.\boldsymbol{u}_j}{|\boldsymbol{u}_i||\boldsymbol{u}_j|}\right), \boldsymbol{u}_i = \boldsymbol{e}_i | \boldsymbol{e}_i.Z = 0 \;\&\; \boldsymbol{u}_j = \boldsymbol{e}_j | \boldsymbol{e}_j.Z = 0 \tag{46}$$

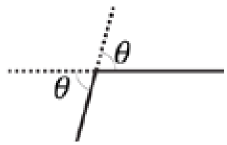


FIGURE 83 shows which angles are considered between streets

In search for a sigmoid function that could accept arguments of type radians, we concluded that squared sine of theta times the arbitrary confusion coefficient τ works consistently as a relative impedance function, besides converting angles measured in Radians to dimensionless numbers. Without loss of generality, we choose an arbitrary amount of 10 seconds for average confusion time in case of maximum change of directionality[a].

This parameter could be calibrated by further empirical research.

$$\zeta^{A}_{(i,j)} := \begin{cases} h(\theta_{(i,j)}), if\ Deg(v_k) > 2 \\ 0\ \ otherwise \end{cases} \mid h(\theta_{(i,j)}) = \tau sin^2 \frac{\theta_{(i,j)}}{2} : \tau = 10\ seconds \qquad (47)$$

$$junction = k \mid a^{ev}_{(i,k)} = a^{ev}_{(j,k)} = 1, i.e. the\ vertex\ coincident\ to\ e_i\ and\ e_j$$

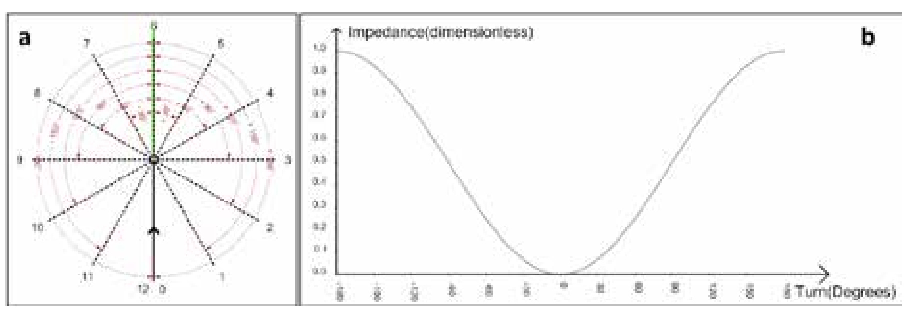


FIGURE 84 Picture (a) shows how the angles are computed for different hypothetical destination streets, given the directed origin street shown in bold black. Note that going from the origin to destination number 6 corresponds to no change in direction and thus zero degrees of turn and no impedance at all. Note also that changing movement direction does not make any difference in computed angles. Picture (b) shows a plot of the turning (cognitive) impedance function as a dimensionless normalized factor.

a To this confusion time, we could potentially add the waiting time corresponding to a traffic light at the same junction. This could be considered as an additional advantage of our method, because this way by adding impedance value for each junction we can potentially avoid dangerous junctions with heavy vehicular traffic as much as possible. To apply this method of path finding in actual path finding scenarios, it would be best to include additional types of impedances for instance to avoid paths that are exposed to high levels of air pollution or noise pollution. Including such other impedances however, requires an extension of the framework towards encompassing various impedances of different physical dimensions. This could be a topic for future research.

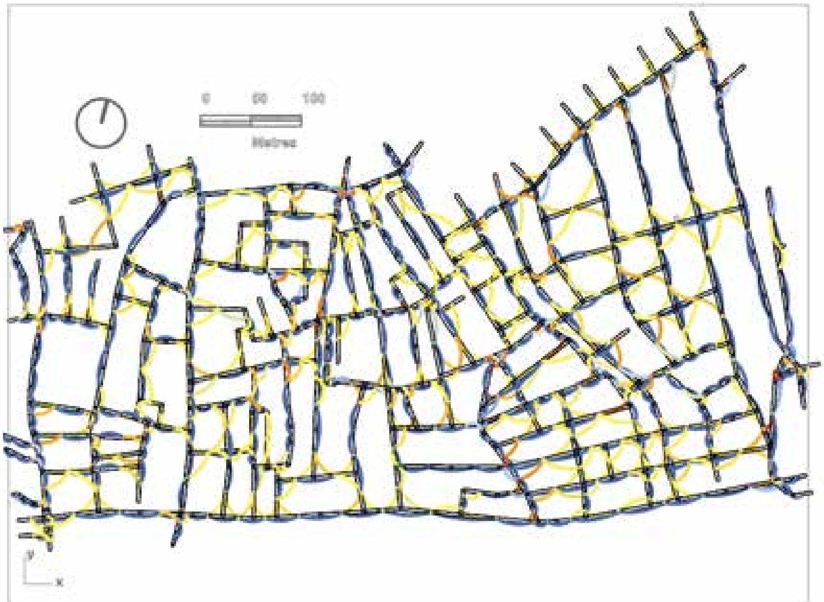


FIGURE 85 shows the angular impedance values attributed to the links between streets

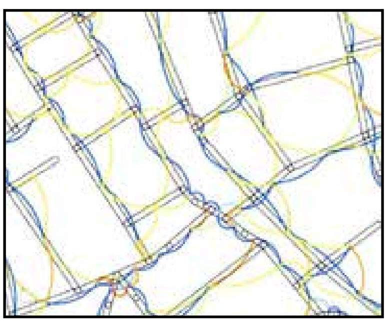


FIGURE 86 shows a close up view of angular impedances attributed as colors to arcs representing the links between streets: the warmer the color the higher the angular impedance.

§ 5.5.2 Easiest Path Algorithm

Here, we give an overview of the Easiest Path as an optimal path problem using the impedance measures introduced above and show examples of such geodesics on an urban network. In our first model of Easiest Path algorithm, we model the street network as a directed graph that has directed street segments (segments of street centrelines) as nodes and their junctions as links. In the second model, we take streets as undirected line segments and represent them as nodes. In either of these graphs, we minimize the impedance of travelling from an origin to a destination. As we have defined both cognitive confusion and physical difficulty in terms of time, they are commensurate and therefore we can use a weighted sum model to model the total impedance of each link. The geodesics are then found using a graph search algorithm. Formally, the algorithm minimizes the total impedance of a path between an origin and a destination (i^{th} node to j^{th} node). A path is defined as a sequence of nodes (i.e. street segments) $\pi = (n_1, n_2, \ldots, n_l) \in N{\times}N{\times}\ldots{\times}N$ such that n_k is adjacent to n_{k+1} for $0 \leq k < l$. The path π is said to be of length l from the first node (n_0) to the last node (n_l). Having defined a real-valued impedance/cost function $Z: L \rightarrow \mathbb{R}$, which attributes an impedance or cost to each link of the graph $\Gamma_d(N, L)$, we need to find a path $\pi = (n_0, n_1, \ldots, n_l)$ that minimizes the total cost or impedance of going from an origin n_o to a destination n_d ($n_o = n_0, n_d = n_l$) over all possible paths between n_o & n_d. Let $L_{i,j}$ be the link in between n_i & n_j, then we need to minimize the following sum (with reference to our prior definitions of impedance): (note that we have denoted the cost function $f(L_{(i,j)}) = \zeta_{(i,j)}$)[61]. In our formulation, we have considered weights of importance for temporal and angular impedances to account for different preferences of tourists or residents.

The first step in computing an Easiest Path is construction of a Street-to-Street adjacency graph, whose nodes are directed or undirected streets. From each edge to another one, there is physical impedance for travelling due to the length of the path and its steepness. The steepness of a path affects the speed of walking or cycling at a normal level of power generation for an average human. Therefore, the slope eventually affects the speed, which based on the length of the path is translated into cost of traveling in terms of time. Note that this cost will be dependent on mode of transportation, i.e. walking or cycling. The other impedance (alias cost) is associated with difficulty of navigation due to demanded change of direction in traversing a street to another. We compute the angle between the direct continuation of a street and the next street and derive a Fuzzy measure of angular impedance that is dimensionless and ranges between 0 and 1, corresponding to no angular change to a full U turn (i.e. 180 degrees of change in movement direction). This is regarded as cognitive impedance of traversing a street to another that is eventually translated into [wasted] time for navigation because of potential confusion in way finding. We do this translation using a parameter dubbed τ (tau), which accounts for the maximum time that a pedestrian/

cyclists would waste at a junction being confused as to which direction is correct as the next step. The assumption validated by previous research is that people (especially tourists and new comers) tend to follow 'their nose' (Dalton, 2003), meaning people prefer simple paths when it comes to navigation. The overall cost of travelling from i^{th} street to j^{th} street is then formulated as follows:

$$\zeta_{i,j} = \begin{cases} \zeta^{W}_{(i,j)} + \zeta^{A}_{(i,j)}, if\ walking \\ \zeta^{C}_{(i,j)} + \zeta^{A}_{(i,j)}, if\ cycling \end{cases} \quad (48)$$

Where $\zeta_{i,j}$ is the cost of going from i^{th} street to j^{th} street; Z denotes physical impedance and is a function of δ(link length); α is the slope of the link in radians; and θ is the planar angle between i^{th} and j^{th} street. The term multiplied by tau represents the cognitive impedance caused by θ and τ represents the amount of confusion attributed to maximum angular change of direction equal to 180 degrees. The typical walking speeds are based on the function defined by Tobler (Tobler, 1993)and cycling speed calculation is based on the work of (Allain, 2013). Walking time and cycling time when traversing i^{th} street to j^{th}street are denoted as $\zeta^{w}_{(i,j)}$ and $\zeta^{c}_{(i,j)}$ respectively. It is notable that these values are parametric and can be adjusted to represent motor assisted bikes. The easiest path is then the path π that minimizes the following sum over all possible paths.

$$\sum_{(i,j)\in\Lambda} Z\left(L_{(i,j)}\right) = \sum_{(i,j)\in\Lambda} \zeta_{(i,j)}\,, \Lambda = \{(i,j)|(i,j) \in L \cap \pi\} \quad (49)$$

Where Z is Greek Zeta, that is the impedance function $Z: L \to \mathbb{R}$ (which assigns a real-valued impedance to each link in the set of graph links L); and Λ (capital Lambda) is the set of links in a 'path' between two nodes[62].

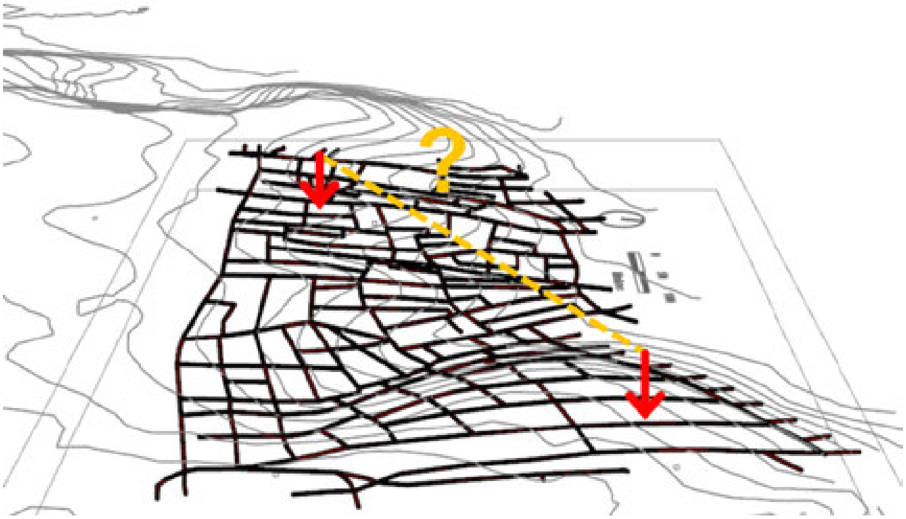


FIGURE 87 the problem of finding a path of minimum length, elevation change, and direction change for everyday commutations can be best understood when shown in 3D.

Solving this minimization problem is done by Floyd-Warshall algorithm to find Easiest Paths between all pairs of origins and destinations. This is a very efficient process with the time complexity of $O(N^3)$ where N is the number of nodes or streets in the network. If only one path is required at a time, then the A* algorithm would be the efficient way of path finding. However, in our analyses, we need all Easiest Paths almost everywhere, therefore we find all using Floyd-Warshall algorithm. As apparent, the innovation of the Easiest Path algorithm is in its problem formulation, and the problem solving part is standard.

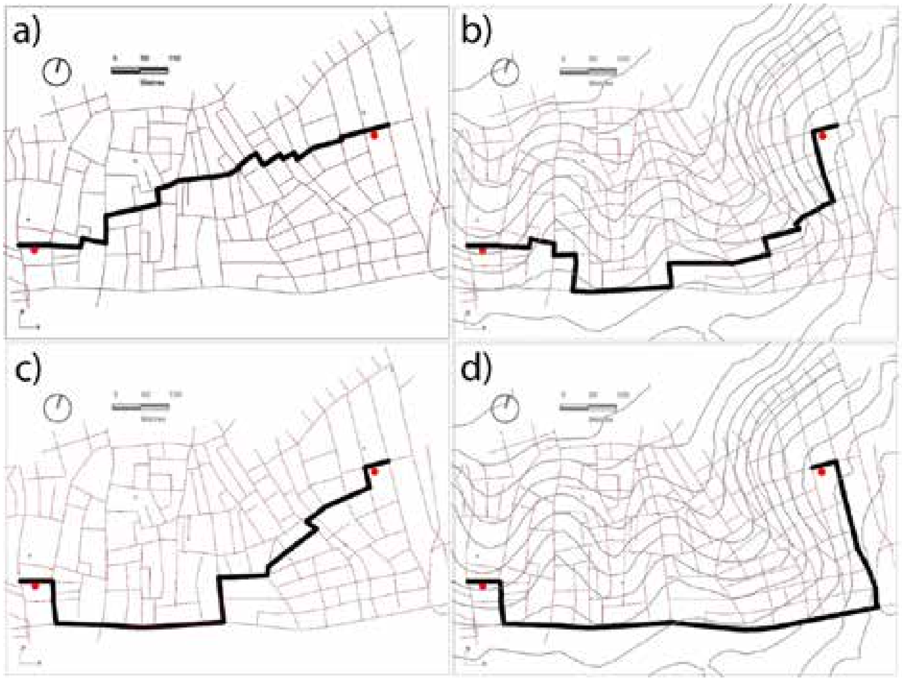


FIGURE 88 shows: a) a Shortest Path without considering the terrain and difficulty of navigation on an example network from "Tarlabasi", Istanbul, data set provided by Ahu Sokmenoglu; b) Easiest Path geodesic found considering the terrain and tau=0 for angular confusion (thereby no cognitive impedance; c) Easiest Path geodesic computed not considering the terrain and tau=15 seconds; d) Easiest Path geodesic computed considering the terrain and tau=15 seconds.

§ 5.5.3 Distance Redefined

We can define Temporal Distance as the time it potentially takes someone to go from O to D through easiest path available. As all costs of travelling are consistently measured in terms of time (minutes), the length of each geodesic [Easiest Path] is the temporal distance as potentially experienced by a pedestrian or cyclists. Note that we have different cost functions for walking and cycling according to the physics of these modes of mobility, which naturally correspond, to smaller temporal distances for cycling. This is a remarkable result as there is no other framework consistently measuring actual temporal distance in one-to-one correspondence with such geodesics.

§ 5.6 Fuzzy Interpretation of Distance Measures as Closeness

Following our experiential direction in research, we go further in modelling distance as perceived regarding the verbal notion of closeness. It is rather obvious that walking or cycling for more than an hour or so is not practical for most people, especially if it is to be a part of their daily routine. This suggest that closeness can be modelled in correspondence with a maximum distance as a threshold above which a person would not be willing to go on foot or bike to a destination, or a distance above which they perceive the destination as absolutely far. We can measure the practicality of walking and cycling as a function of walking or cycling temporal distance, given a threshold that show 'how far' (denoted as F) a person is/might be potentially willing to go on foot/bike. Fuzzy variables can range in between 0 and 1 therefore we need a function that can map distance values ranging from 0 to +∞ to values between 0 and 1. Inspired by Logit models in discrete choice models of transportation forecasting models, we choose a Logistic Function as below, which represents the degree to which a statement such as 'destination D whose distance to origin O is x is close by' is regarded as true. Another way of interpreting this measure would be as utility or suitability of walking or cycling as a mode of transportation given a temporal travel distance.

$$C(x) = \frac{1}{1 + e^{\mu\left(x - \frac{F}{2}\right)}} \quad (50)$$

In this equation, C(x) denotes closeness of a destination at a distance x; and λ represents a coefficient whose role is to ensure the decline of the closeness value when distance x approaches F (i.e. the furthest temporal distance a person is willing to go on foot or by bike). Note that the alternative crisp logic representation would be that: all destinations farther than F would have been regarded as far; and those closer than F would have been regarded as close. Thus, the advantage of this Fuzzy representation should be apparent.

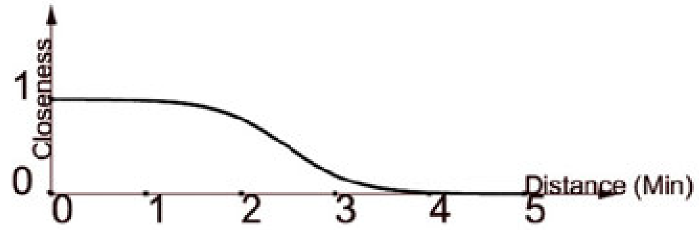


FIGURE 89 shows a plot of the Fuzzy model of closeness given a 'how far' parameter equal to 5 minutes.

We need to find an appropriate value for μ to ensure that the value of closeness goes below a threshold ε at the distance of F (far); it would be best to define λ as a function of this threshold and the distance threshold F:

$$C(x) = \frac{1}{1+e^{\mu(x-\frac{F}{2})}} \le \varepsilon \xrightarrow{yields} 1 \le \varepsilon(1+e^{\mu(x-\frac{F}{2})}) \xrightarrow{yields} \frac{1}{\varepsilon}-1 \le e^{\mu(x-\frac{F}{2})} \quad (51)$$

$$x = F \xrightarrow{yields} \frac{1}{\varepsilon}-1 \le e^{\mu(\frac{F}{2})} \xrightarrow{yields} Ln(\frac{1}{\varepsilon}-1) \le \mu(\frac{F}{2}) \xrightarrow{yields} \boxed{\frac{2}{F}Ln(\frac{1}{\varepsilon}-1) \le \mu} \quad (52)$$

FIGURE 90 shows from left to right the network distance (through Easiest Paths, Cycling, Tau=20 seconds) as conventionally visualized using thermostat colours, 5 minutes catchment of the point highlighted in blue, and the fuzzified distance shown in tones of turquoise. Note that after the distance is fuzzified all distances above the threshold (how far one is prepared to go on bike) are considered practically as infinity, therefore their Fuzzy closeness is zero.

As visible in the figure above, the fuzzified closeness value exactly represents our intuitive understanding of closeness having in mind our means of transportation. For example, when travelling by bike, a destination more than e.g. 1 hour away from an origin is absolutely far for many people.

We can project any measurement performed on network segments to 2D features (urban plots) that are accessed through network segments. This is done by attributing the measurements pertained to line segment of the network that is closest to the plot in question. Of course, the access point of some plots/parcels can be slightly different in some cases and this attribution can be done in some other way that reflects the reality more accurately, provided there is a detailed survey available that clarifies which plot is accessed exactly through which segment.

FIGURE 91 shows Fuzzy closeness for cycling from the origin marked (as blue dot considering the terrain, tau=30 seconds, the sharper the colour the closer the destination. Note that the distance values have been projected to the urban plots assuming that each plot is accessed through the closest street segment.

§ 5.7 Fuzzy Accessibility Measures for Pedestrians and Cyclists

Here we give two fuzzy definitions of closeness that plainly model feasibility of accessing destinations of interest given the time people are willing to spend walking or cycling towards them. Suppose for example, there are four grocery stores in a neighbourhood, but some of them are more favourable so people are willing to go somewhat farther on foot/bike to get to them. Such preferences can be modelled by attributing a number to each point of interest (POI) saying how far one would be willing to go on foot or bike to get there.

§ 5.7.1 Proximity (closeness to all POI)

The 'Proximity to All' (Proximity in short) tells how close a location to all destinations of interest is. It thus tells whether all interesting locations (attractions) are accessible given abovementioned willingness (how far) parameters. If this measure is computed for all possible destinations as potential destinations, it will generate a local closeness centrality measure comparable with local integration in Space Syntax. A number of advantages compared to 'local integration' can be listed as follows: that our measure of local closeness centrality can work for any number of desired destinations; that its meaning is physically tangible, i.e. does not require pages of explanation; and that it can be interpreted as temporal accessibility as experienced. It simply tells to what extent it would be true to consider all locations (or some locations) as close to an origin, given the maximum distance above which a destination is considered far away. This measure is computed using Fuzzy AND operators on fuzzified closeness values.

§ 5.7.2 Vicinity (closeness to any POI)

The 'Vicinity of Any' (Vicinity in short) tells how close a location to any destination of interest is. It thus tells whether any of interesting locations (attractions) is accessible given abovementioned willingness (how far) parameters. This measure is interesting as it can reveal the polycentric nature of a neighbourhood given a number of comparably interesting attraction places. More simply, a very straightforward application of this measure is to see whether for instance each location has a reasonable access to a grocery store by walking or cycling. This is important because then such daily routine trips can be made without using personal cars. This measure is computed using Fuzzy OR operators on fuzzified closeness values.

§ 5.7.3 Fuzzy framework of CONFIGRAPHIX

We have generalized three types of dyadic Fuzzy operators for multiple inputs and implemented them in our closeness measurement method: namely, Zadeh (Zadeh, 1965), Yager (Yager, 1980), and Paraboloid AND & OR Fuzzy logical operators as shown in the figure below.

$$Zadeh\ AND: \bigcap_i \{x_i\} \coloneqq \min_i\{x_i\} \tag{53}$$

$$Zadeh\ OR: \bigcup_i \{x_i\} \coloneqq \max_i\{x_i\} \tag{54}$$

$$Yager\ AND: \bigcap_i \{x_i\} \coloneqq 1 - \min\{\left(\sum_i (1-x_i)^p\right)^{\frac{1}{p}}, 1\} | p \geq 1 \tag{55}$$

$$Yager\ OR: \bigcup_i \{x_i\} \coloneqq min\{\left(\sum_i {x_i}^p\right)^{\frac{1}{p}}, 1\} | p \geq 1 \tag{56}$$

$$Paraboloid\ AND: \bigcap_i \{x_i\} \coloneqq \prod_i (x_i)^p\ | p \in (0,1] \tag{57}$$

$$Paraboloid\ OR: \bigcup_i \{x_i\} \coloneqq 1 - \prod_i (1-x_i)^p\ | p \in (0,1] \tag{58}$$

In the test data set used as a case study, the numeric differences between different Fuzzy aggregators (alias connectives, speciality in Yager's terms) are negligible or inconspicuous when visualized. Nevertheless, we have implemented and provided all methods to let empirical researchers find the best in their practice.

In spite of their apparent simplicity, the original aggregators defined by Zadeh, seem to make most sense in our context, in that they are easy to understand, interpret, and explain. For example, if your minimum closeness (corresponding to maximum distance) to POIs is 0.2, then the statement "you are close to all of them" is 20% true. This statement corresponds to the fact that from your location the farthest POI is rather far. Of course such fuzzified values can be defuzzified (i.e. traced back to their origins), so as to say to what distance such a fuzzy value correspond. It is interesting to observe that the fuzzified distance (fuzzy closeness) can be interpreted as the probability that a person chooses to walk or cycle to their destination as they find it close. For defuzzification we need to find the inverse of fuzzification function.

We first rewrite the Logistic function that we used for fuzzification:

$$C(x) = \frac{1}{1 + e^{\lambda(x - \frac{F}{2})}} = \frac{1}{1 + e^{-\lambda(\frac{F}{2} - x)}} \tag{59}$$

By setting an auxiliary variable $y = \lambda(\frac{F}{2}\text{-}x)$ we can rewrite the function as:

$$f(y) = \frac{1}{1 + e^{-y}} \tag{60}$$

Now, we need to find an inverse function that we dub $g(y)$:

$$g(y) = f^{-1}(x)\,|f \circ g(y) = id_y = y \tag{61}$$

We need to find y in terms of $f(y)$:

$$f(y) + e^{-y} f(y) = 1 \tag{62}$$

$$e^{-y} = \frac{1 - f(y)}{f(y)} \tag{63}$$

$$e^{y} = \frac{1 - f(y)}{f(y)} \tag{64}$$

Back to our interpretation of the logistic function as the probability of making a choice as to walk or cycle depending on the distance of a destination, we can see that the above equation actually describes the 'odds' of making such a choice, that is the probability of making such a choice over the probability of not making that choice. For this reason the following function is called **Logit** (logarithm of odds):

$$\log(e^{y}) = \log\left(\frac{1 - f(y)}{f(y)}\right) = \text{logit}(f(y)) \tag{65}$$

$$y = \log\left(\frac{1 - f(y)}{f(y)}\right) \tag{66}$$

We assume $p = f(y)$ as the argument of the function defined above, and then it is clear that the following is the inverse function we were after. i.e.:

$$g(p) = \log\left(\frac{1 - p}{p}\right) \tag{67}$$

Because: $f \circ g(y) = id_y = y$.

Therefore, in order to defuzzifiy a fuzzified distance (a closeness value) we can apply this inverse function first:

$$g\big(C(x)\big) = \log\left(\frac{1 - C(x)}{C(x)}\right) = y \quad (68)$$

Now we just need to find the actual distance x in terms of y, given:

$$y = \lambda(\frac{F}{2} - x) \quad (69)$$

$$y = \lambda\frac{F}{2} - \lambda x \quad (70)$$

$$x = \frac{y - \lambda\frac{F}{2}}{\lambda} \quad (71)$$

$$\boxed{x = \frac{\log\left(\frac{1 - C(x)}{C(x)}\right) - \lambda\frac{F}{2}}{\lambda}} \quad (72)$$

$$\boxed{x = \frac{\operatorname{logit}(C(x)) - \lambda\frac{F}{2}}{\lambda}} \quad (73)$$

In case $C(x)$ is a fuzzy aggregate (Zadeh, Yager or Paraboloid) then a representative value of F (farness threshold) to defuzzify the aggregate closeness values. For example, in case of Zadeh operators, minimum or maximum of the farness thresholds would be appropriate as representative farness thresholds. For a better understanding of how the Fuzzy aggregation operators work, see Figure 92, Figure 93, Figure 94, and Figure 95.

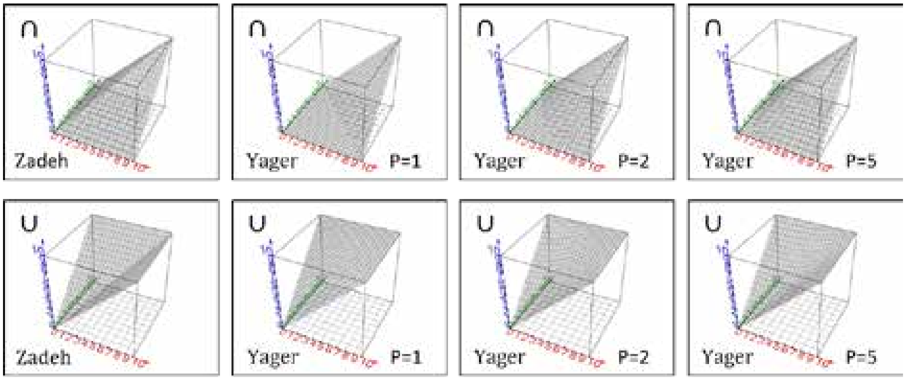


FIGURE 92 a comparison of the generalized Yager AND and OR aggregators with those of Zadeh for 2D inputs along X and Y axes, outputs are visualized as a mesh coloured from black to turquoise as for 0 to 1.

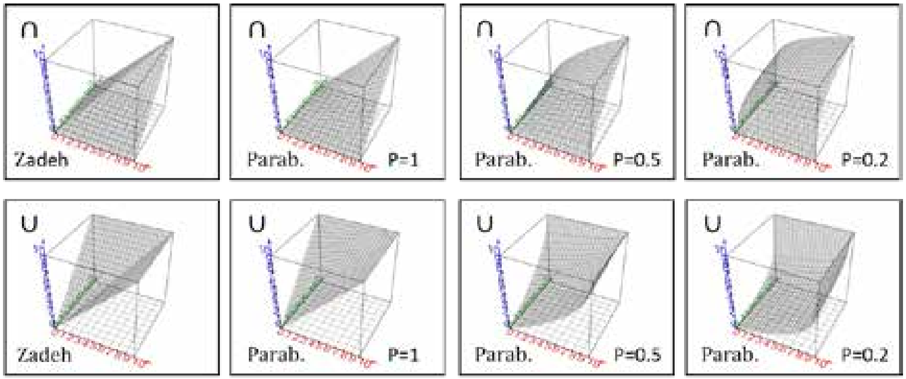


FIGURE 93 a comparison of the generalized Paraboloid AND and OR aggregators with those of Zadeh for 2D inputs along X and Y axes, outputs are visualized as an interpolated mesh.

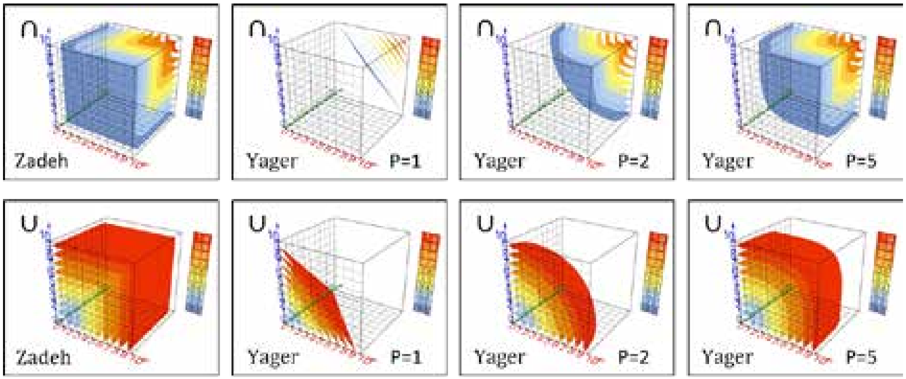


FIGURE 94 a comparison of the generalized Yager AND and OR aggregators with those of Zadeh for 3D inputs, level sets visualized using Marching Cubes algorithm, inputs come along X,Y, and Z axes, outputs are coloured from blue to red as for 0 to 1.

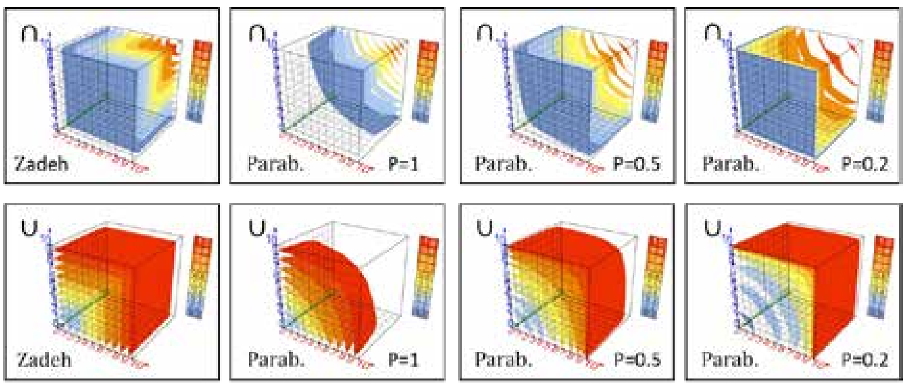


FIGURE 95 a comparison of the generalized Paraboloid AND and OR aggregators with those of Zadeh for 3D inputs, level sets visualized using Marching Cubes algorithm, inputs come along X,Y, and Z axes, outputs are coloured from blue to red as for 0 to 1.

FIGURE 96 shows Closeness to Any POI (vicinity) computed using Zadeh Fuzzy operator

FIGURE 97 shows Closeness to Any POI (vicinity) computed using Zadeh Fuzzy operator projected to plots

FIGURE 98 shows Closeness to All POI (vicinity) computed using Zadeh Fuzzy operator

FIGURE 99 shows Closeness to All POI (vicinity) computed using Zadeh Fuzzy operator projected to plots

§ 5.8 Catchment as Crisp Closeness

If a simple yes or no answer to questions such as the following are needed then the catchment measure (to all/to any) can be used.

- Are all interesting destinations accessible within 5 minutes walking from here?
- Is any of interesting destinations accessible within 5 minutes walking from here?

Note that the catchment measure proposed here is different from conventional alternatives in that it is polycentric; can be computed to all or any of POI; and that it is based on preferred 'how far' parameters. The catchment measures are computed by treating the fuzzy closeness measures as crisp closeness measures.

FIGURE 100 a) shows proximity catchment (to all POI), walking, considering the terrain and tau=15; b) shows vicinity catchment of POI (access to any POI), walking, considering the terrain when tau=15

§ 5.9 Zoning for Facility Location and Business Intelligence

Looking at the catchment analysis results, we asked ourselves whether it is possible to tell to which POI each location has preferred access. To answer this question we modelled generalized alpha shapes and Voronoi diagrams (Edelsbrunner, H., & Harer, J., 2008) to divide the network space to areas of preferred POI. This is closely related to vicinity and vicinity catchment (closeness to any POI). It adds a new dimension to the analysis by specifying how the POI serve/take shares of a neighbourhood considering walking/cycling access. We provide two forms of this measure that we call inclusive and exclusive zoning. The former gives an answer to 'which POI is preferred' regardless of whether it is accessible within the acceptable range of distance or not; whereas the latter excludes locations that are by no means accessible (considering the acceptable ranges of distance as specified by user). Generalizing the definition of Power of a Circle over a Point in Euclidean geometry given as $h(D,R) \equiv D^2\text{-}R^2$, where D denotes the distance of the point –via Easiest Paths- to the circle centre and R the radius of the circle in question we define:

$$InclusiveZone(n_i) := \chi \equiv \underset{i}{arg\,min}\, Power\,(i,j) = D^2_{(i,j)}\text{-}R^2_j \qquad (74)$$

$$ExclusiveZone(n_i) := \begin{cases} \chi \equiv \underset{i}{arg\,min}\, Power\,(i,j), if\ D_{(i,j)} < R_\chi \\ \text{-}1, otherwise\ (undefined) \end{cases} \qquad (75)$$

The power of a point relative to a circle of radius R is positive outside the circle, zero on the circle, and negative inside the circle[a]. Thinking of the definition of a circle in Euclidean geometry as the locus of points whose distance to a point is equal to a constant radius, we see that a circle can also be conceived in our network space where radial straight lines (Euclidean geodesics) have been replaced by radial geodesics found by Easiest Path algorithm. The rest of the concepts generalize similarly. Using the concept of power, we can account for the fact that different points of interest might have different suitability or attraction for dividing the neighbourhood into zones. Using a simplified version of zoning, as finding the closest POI for each point would not allow for this consideration. As an example, consider a few grocery stores that provide similar services and thus it is fine for an inhabitant of the neighbourhood to be close by **any** of them; but most people find one store more attractive than others and therefore they would not mind walking or cycling an extra 2 minutes towards it. See Figure 101, Figure 102, and Figure 103. The left images show inclusive zoning where radii of access have been considered as weight of influence for producing a generalized Voroni diagram. The right images show exclusive zoning where radii of catchment have been intersected with Voronoi cells; this corresponds to a generalized alpha shape. The zoning is shaped according to the How-Far parameters set to 4, 2, 4, and 4 minutes corresponding to POI.

a For more information on the concept of power of a point relative to a circle look at: http://mathworld.wolfram.com/CirclePower.html

FIGURE 101 shows cycling zones of the neighbourhood with regard to preferred POI within arbitrary radii of catchment.

FIGURE 102 shows walking zones (left: inclusive, right: exclusive) of the neighbourhood with regard to preferred POI within arbitrary radii of catchment.

FIGURE 103 shows an alpha shape diagram computed using Euclidean distance metric from the same POI with the same radii of access. Although closely related, using a diagram with Euclidean distance as metric can be misleading in assuming better access (farther reach).

§ 5.10 On Network Centrality Models

It is interesting for planning professionals to know what would be the effect of a change in the network in terms of the pedestrian or cyclist flows. Medial (accounting for paths or walks passing through a node) or Radial (accounting for paths or walks passing to a node) centrality models[63] can help us see two different aspects of networks more clearly, regarding how a network structures or biases "movement to" or "movement through" certain spaces[a].

Network Centrality models can be useful in identifying most important nodes by means of a ranking mechanism. There are different centrality measures used for different purposes and different types of networks. Centrality models should not be mistaken with performance models. They often reveal a structural property of the underlying graph, but making a performative sense of such properties requires an extra layer of assumptions that could relate the structural properties to the dynamics of a network.

a terminology of Space Syntax from (Hillier, 2007)

It is notable that most of the well-known centrality measures have been originated within the field of Social Network Analysis. Regarding our point of departure, i.e. studying the "social logic of space", this is a natural choice for studying the collective/social mobility behaviour. Note that this field has emerged many years before the widespread availability of online social networks on World Wide Web (prominent models had been made in 1950's, 60's and 70's). We have adapted a few centrality models from the field of social network analysis to our methodology by means of using the geodesics, distances or fuzzified distances computed with Easiest Paths. The first class uses directly the geodesics or geodesic distances, the second class, which we call spectral (as in Spectral Graph Theory) deals with the eigenvalues and eigenvectors of the matrices pertained to the underlying network (Adjacency Matrix, Laplacian Matrix, or Transition Probability Matrix of a Markov Chain). The results of Geodesic centrality models can be understood intuitively as they are almost self-explanatory. The spectral centrality measures pave the way for introducing probabilistic models of walking/cycling mobility.

§ 5.11 Geodesic Centrality Models

Space Syntax indicators such as Integration (a variant of closeness centrality) and Choice (betweenness centrality) are both defined based on different notions of 'geodesics' or shortest paths, be it topological shortest path or angular shortest path. We can reconstruct such indicators by substituting these geodesics with Easiest Path for network studies focused on walking and cycling. Here we focus on betweenness centrality as it can be interpreted as the probability that people would pass through a certain street going from some place to another, considering that they are moving through easiest paths. This is of course of importance for retail businesses as they depend on the probability of passage of pedestrians. Similar to all other indicators, in computation of centrality measures we can opt for different values of confusion time for cognitive impedances to take account of preferences of people who care more for shortness of routes or those who prefer simpler routes.

§ 5.11.1 Betweenness, Local Betweenness

The measure of 'betweenness' was introduced by Linton Freeman as an indicator of importance of nodes in a social network (Freeman, 1977). Considering all shortest paths in a network between all possible pairs of nodes, we can find out how often a node happens to be on a shortest path between two other nodes. We can interpret this as the probability that a person passes through a certain street segment given all other possibilities. Variations of this model based on angular shortest paths or metric shortest paths have been found to have high correlations with the location of businesses such as retail, cafes, and pubs (examples mentioned in (Hillier, 2007) & (Sevtsuk, 2010)). The new element in our model is the geodesics we have introduced as the most convenient paths for pedestrians and cyclists. As is the case with any model, our model is based on a reduction of complex reality so it can never perfectly explain how people find some routes more convenient (or pleasant) and actually take them from their origins to their destinations. The reality is of course more complicated due to many other decision variables, many of which are perhaps related to the functional aspects of urban trips.

The probability interpretation leads us to consider dividing interesting possibilities by the total number of possibilities; therefore, we divide the total number of geodesics that include the node in question by the total number of all geodesics. Given that the graph is connected, then the total number of geodesics equals the total number of pairs of origin-destination. This corresponds to the number of combinations of two nodes from all nodes, excluding the node in question. We can define the bare probability of passage of a body through a node (in absence of attractions and other information) as in the equations below, in which γ_{st} is the geodesic path between source s and target t and $\sigma(s, n_i, t)$ is a binary variable that is equal to one if the geodesic γ_{st} (i.e. a sequence of nodes) contains the node in question (n_i). Note that this is a simplification of the original Betweenness Centrality as defined by Freeman; because we have assumed it is very unlikely that there exist more than one geodesic in between a pair of nodes. A similar simplification, for a similar reason is used in (Turner, 2007).

$$|\{(s,t)|s \in N, t \in N, s \neq i \neq t\}| = \binom{|N|\text{-}1}{2} = \frac{(|N|\text{-}2)\times(|N|\text{-}1)}{2} \quad (76)$$

$$\boldsymbol{c}^B(i) = \frac{2\times \sum_{s=1}^{|N|}\sum_{t=1}^{|N|}\sigma(s,i,t)}{(|N|\text{-}2)\times(|N|\text{-}1)} \mid s \neq i \neq t\,, \sigma(s,i,t) = \begin{cases}1, if\ \ \gamma_{st} \ni i\\ 0, otherwise\end{cases} \quad (77)$$

Betweenness centrality literally shows how often a street happens to be on an Easiest Path between an origin and a destination. It is notable that we have revisited the concept of “local choice” (betweenness in Space Syntax jargon) and made it possible to compute betweenness for a temporal range of distance. We can also compute ‘local betweenness’ to find out which streets are most likely to be traversed in trips shorter than e.g. 5 minutes. As it is the case with any kind of betweenness centrality measure, they essentially look at purposeful trips between origins and destinations but not wandering and lingering. See Figure 104, Figure 105, Figure 106, Figure 107, and Figure 108 for exemplary results.

FIGURE 104 shows betweenness centrality computed considering all Easiest Paths in the network

FIGURE 105 shows the effect of tau, the confusion-time parameter on the betweenness values computed. Note that when tau is put to zero, the easiest paths do not favour straighter paths and therefore the centre of the neighbourhood necessarily is highlighted. However, when tau is set to higher values then straightest or the most important routes are more clearly distinguished from the rest.

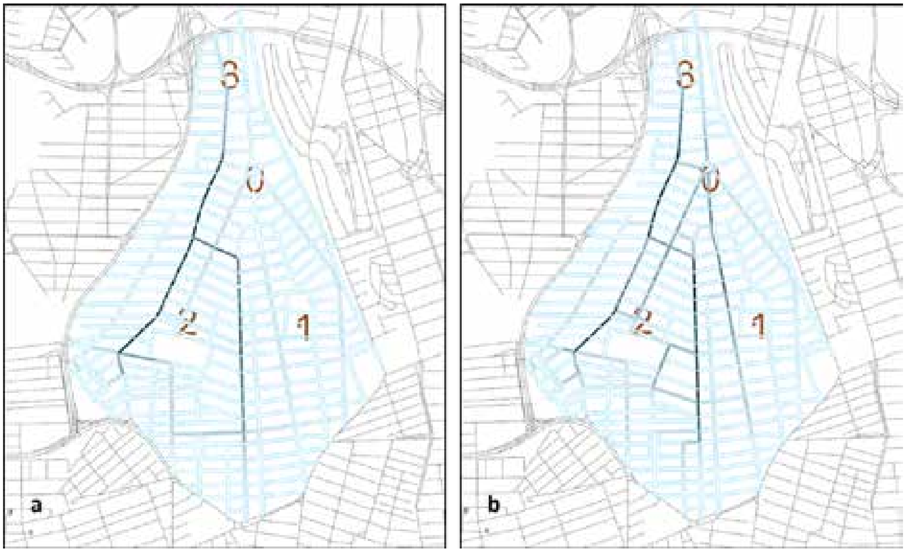


FIGURE 106 (a) shows the betweenness centrality when the geodesic is only angular and the weight of physical distance is zero; and (b) shows betweenness centrality when both angular and temporal impedances have been given equal weight. It is visible that the picture (b) takes better account of reality as to importance of main roads of the neighbourhood have been revealed better compared to the case (a) when the algorithms disregards the physical distance.

FIGURE 107 shows betweenness centrality values computed for Morwell, Asutralia, at the radius of 10 minutes walking. There was not a terrain model available, tau is set to 15 seconds for confusion time.

FIGURE 108 shows betweenness centrality values computed for Morwell, Asutralia, at the radius of 4minutes walking. There was not a terrain model available, tau is set to 15 seconds for confusion time.

§ 5.11.2 Closeness, Local Closeness

Closeness centrality for a node (street space in this case) is defined as the inverse of its average Easiest Path distance to all other nodes (global closeness) or nodes (streets) closer than a certain radius (local closeness). Formulation of the local closeness measure is inspired by Local Integration in Space Syntax. With a similar purpose, we can look at local closeness centrality as an indicator of the potential of a node to be a common destination or an attractor of "movement to" itself.

$$\boldsymbol{c}^{C}(n_i) = \frac{|P_i^R|}{\sum_{j \in P_i^R} \boldsymbol{D}(n_i, n_j)} \, |\boldsymbol{D}(n_i, n_j) = \sum_{k \in \gamma_{n_i, n_j}} \zeta_k \;\&\; P_i^R = \{n_j | D(n_i, n_j) < R\} \quad (78)$$

The interesting fact about this centrality measure is that it can reveal the polycentric structure of a neighbourhood at various radii of access, and ultimately for different modes of transportation.

This is to say this method can be used in a way rather opposite to starting with points of interests; it helps us find potential points of interests at the local maxima of local centrality function. In order to find these maxima, we need to find nodes whose closeness values are above all their neighbours. This is done through the following method:

$$Maxima = \{n_i | C_i > C_j \; \forall j | D(n_i, n_j) < R\} \quad (79)$$

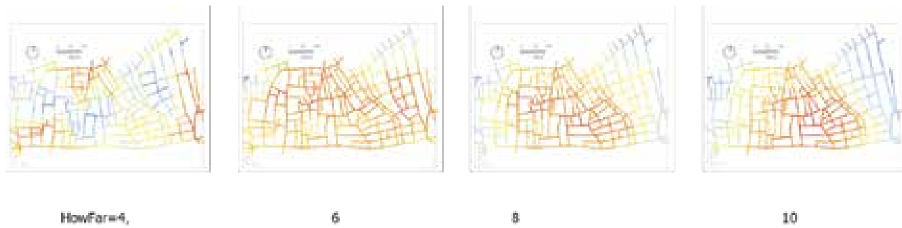


FIGURE 109 shows local closeness centrality at different radii.

FIGURE 110 shows a monocentric structure with the [almost] global maximum highlighted in black for closeness centrality at the radius of 10 minutes cycling.

FIGURE 111 shows the polycentric structure corresponding to the local maxima of local closeness centrality computed at the radius of 4 minutes cycling.

FIGURE 112 local closeness on Morwell, Australia, cycling radius 15 minutes.

FIGURE 113 local closeness on Morwell, Australia, cycling radius 10 minutes.

§ 5.12 Spectral Centrality Measures on Configuration Graphs

The term spectral refers to a notional spectrum corresponding to an eigenvector-eigenvalue decomposition of a matrix; i.e. the set of eigenvalue-eigenvector pairs of a graph matrix. Spectral graph theory does not seem very intuitive on the surface. To give a background to the reader, we begin this part by introducing Spectral Graph Drawing as intuitive evidences as to why graph spectra are important and what they tell about the structure of a graph. We can think of three main types of matrices associated with graphs, namely: Adjacency Matrices, Laplacian Matrices, and Transition Probability Matrices of Markov Chains (simulating Random Walks). From Linear Algebra, we know that a square matrix can have eigenvectors and eigenvalues, which are special directions along which the underlying transform represented with the matrix acts linearly. This eventually means that any state/position/vector in $\mathbb{R}^N$ (i.e. a Hilbert Space) can be represented as a linear interpolation of eigenvectors; and as the matrix works as a linear transform in such directions then all linear transformations can be written as combinations of linear transformations. Spectral Graph Drawing, i.e. embedding a graph in a low-dimensional Euclidean space with Cartesian coordinates obtained from its eigenvectors was initially proposed using Adjacency Matrices of graphs, and later using Laplacian matrices. This latter formulation is very common and dates back to 70's (Hall, 1970). We have additionally adopted a newer elegant formulation, which utilizes a third category of matrices that can be thought of as Transition Probability matrices of some Markov Chains on graphs (Koren, 2003).

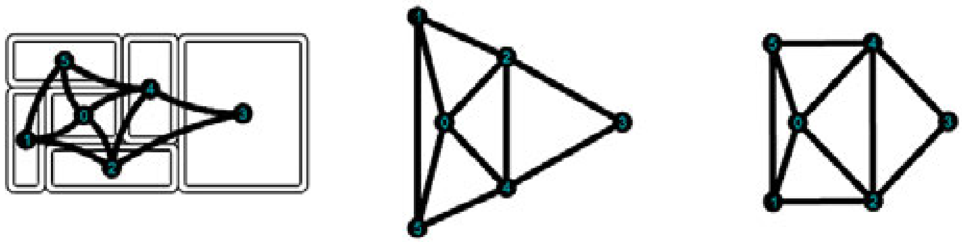


FIGURE 114 spectral graph drawing of a cell-configuration graph at left, using Laplacian matrix (middle) and using the Kroen's algorithm (right)

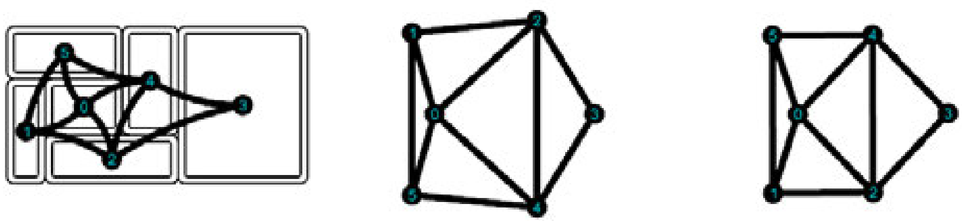


FIGURE 115 spectral graph drawing of a cell-configuration graph at left, using Symmetrically Normalized Laplacian matrix (middle) and using the Kroen's algorithm (right)

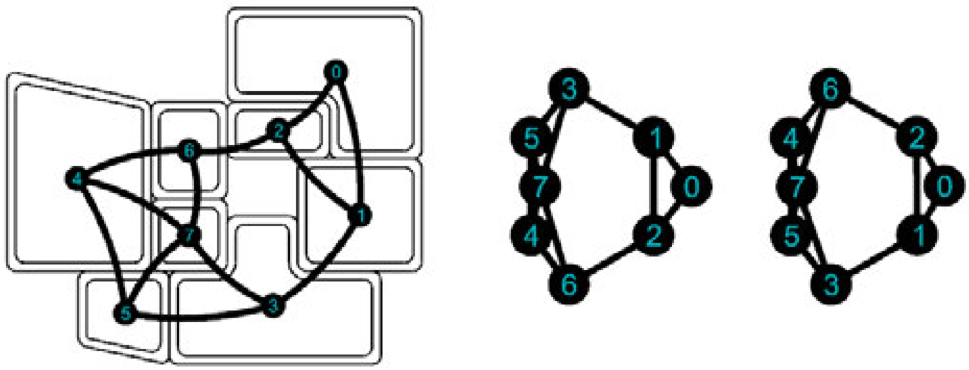


FIGURE 116 spectral graph drawing of a cell-configuration graph at left, using Laplacian matrix (middle) and using the Kroen's algorithm (right)

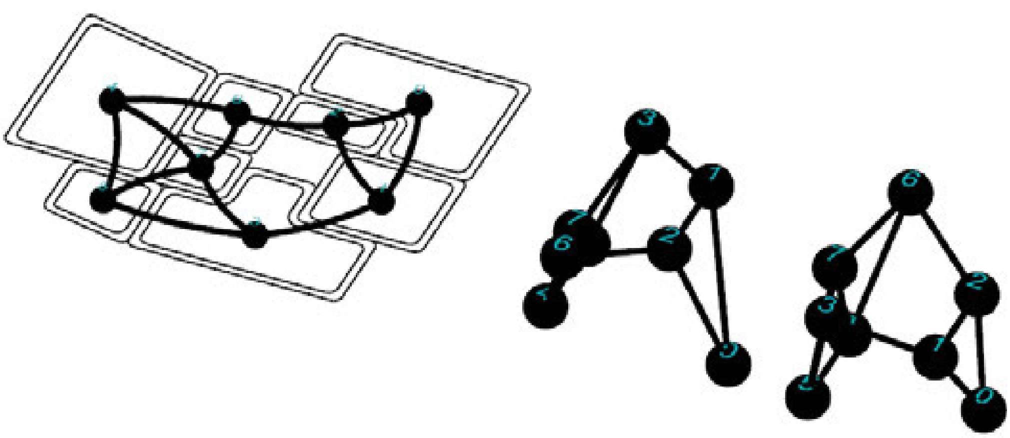


FIGURE 117 shows the same configuration when embedded in 3D Euclidean space. Observe the differences of the two methods (Laplacian and Kroen's method)

FIGURE 118 shows a 3D spectral drawing of the dual-undirected graph of Tarlabasi neighbourhood in Istanbul.

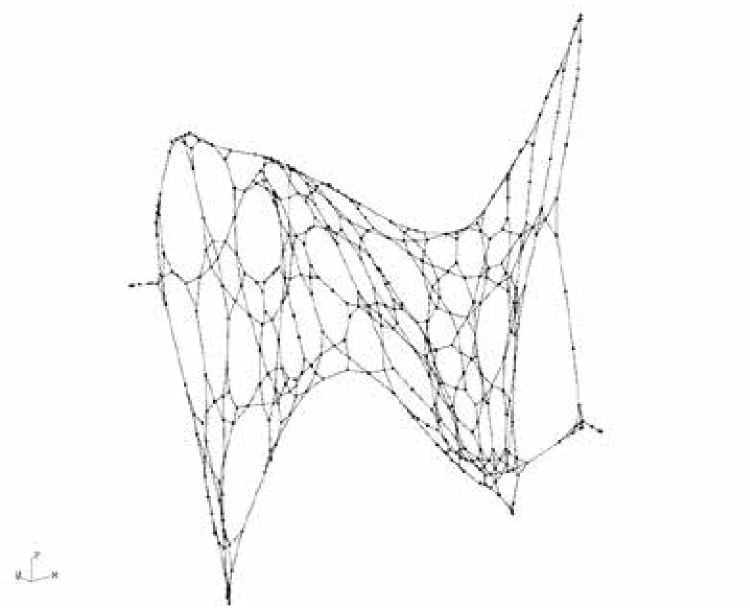


FIGURE 119 shows a degree-normalized spectral drawing of Tarlabasi dual graph network of streets, using our implementation of Kroen's algorithm. This one clearly is more similar to the map and better resolved.

In simple terms, the idea of Spectral Graph Drawing is to embed (assign a geometric location to the abstract vertices of) a graph in 1D, 2D, or 3D Euclidean space using the non-trivial eigenvectors of a matrix associated with this graph. The brilliant point about a spectral drawing is that it is unique and that it is an exact solution to the problem of graph drawing. This is what distinguishes spectral drawing from all other solutions to the problem. Yet, spectral drawing according to the definition we use is tightly related to the intuitive Force-Directed Graph Drawing algorithm that we introduced in Chapter 3. The point we want to make is that a spectral drawing of a graph can be seen as its fingerprint. In addition to providing deep insight into the idea of graphs as abstract entities, the topic serves as an introductory bridge to the remainder of this part that is on Spectral (a.k.a. degree) Centrality measures.

§ 5.12.1 A Very Short Introduction to Spectral Graph Theory

Spectral Graph Theory studies properties of graphs by inspecting the eigenvalues and eigenvectors of the matrices associated with graphs, namely the Adjacency Matrix, the Laplacian Matrix, and the Transition Probability Matrix (pertained to a Markov Chain). We have already introduced the Adjacency Matrix and we will introduce the Transition Probability Matrix in the context of our probabilistic models. Here we mainly focus on the Laplacian Matrix and its spectrum, i.e. its eigenvalues and eigenvectors.

This part gives a very brief overview of the most essential topics in spectral graph theory, which is a subject that is not so easy to understand with pure intuition, at least in its current state of representation. Of course, this summary cannot cover the topics in depth and it does not contain any proofs. It should be viewed as a storyline that is to shed some light on a number of underlying concepts in Spectral Graph Theory[64]. We, however try to give a quick guide to the fundamentals that are necessary for defining the rest of the centrality measures in this chapter.

Laplacian matrix performs an operation on functions defined in the Hilbert space ($\mathbb{R}^N$) denoted by the graph that is (related to) the discrete analogous of the Laplacian operator (Del or nabla operator shown as ∇) in continuous calculus (as used in gradient, divergence, and curl definitions), i.e. measuring the differentials of a function along different directions. Suppose a vector-function (N-dimensional array) $\mathbf{f} \in \mathbb{R}^N$ that assigns a real value to every node in the graph. As an example, say a number representing noise-level or number of pedestrians measured on every street. If we want to find out to what degree this function varies at each node in comparison to neighbour nodes then we can use the Laplacian matrix.

There is also a product known as a 'quadratic form' associated with the Laplacian that represents a very important distribution:

$$\boldsymbol{f}^T \boldsymbol{L} \boldsymbol{f} = \sum_{(i,j)\in L} \left(\boldsymbol{f}(i) - \boldsymbol{f}(j)\right)^2 \tag{80}$$

This can be intuited as the sum of generalized Euclidean distances between the values of the function in question in the Hilbert space (the discrete space defined on the graph). More clearly, if we attribute to each vertex a position in the Euclidean space of $\mathbb{R}^3$ this quadratic form would be proportionate to the amount of potential energy in an imaginary system composed of ideal springs put for links. This is closely related to the idea of drawing a graph using our force-directed graph-drawing algorithm. Remember that the amount of potential energy in a spring that is not relax, i.e. has displacement of X from relax position is equal to $\frac{1}{2}\mathrm{kx}^2$. Putting $\mathrm{x} \in \mathbb{R}^3$ instead of f would then translate the quadratic form to $\mathbf{x}^{\mathrm{T}}\mathbf{L}\mathbf{x} = \sum_{(\mathrm{i,j})\in \mathrm{L}} (\mathbf{x}(\mathrm{i}) - \mathbf{x}(\mathrm{j}))^2$. Then it is clear that this can be seen as the sum of potential energy in the springs representing the links in our graph. It is obvious that minimizing this sum would bring the system towards and equilibrium state where neighbours are positioned close to each other. To achieve this as a non-degenerate solution of course we need to make sure that the vertices representing the nodes of our graph do not fall onto each other. Mathematically, we need to ensure the variance of the vertex positions is non-zero.

The variance can be formulated as: $\frac{1}{\mathrm{n}}\sum_{\mathrm{i}} (\mathbf{x}_{\mathrm{i}} - \mu)^2$, where μ denotes the average position off vertices. By setting $\mu = \mathbf{0}$ then the variance is equal to:

$$\frac{1}{n}\sum_{i} (x_i)^2 = x^T x \tag{81}$$

Therefore, the energy minimization problem associated with the force-directed drawing algorithm can be seen as a constrained optimization problem or minimization of the '**Rayleigh-Ritz Quotient**' with respect to the Laplacian that is exactly the above-mentioned "quadratic form" representing the sum of squared differences of positions of values of nodes across the links. The Laplacian Matrix is Hermitian; that is its conjugate transpose equals itself, which is necessarily true as the Laplacian matrix is only consisted of real-valued entries and that it is symmetric because the underlying graph in our case is undirected. This entails a number of nice properties as follows: the eigenvalues of the Laplacian matrix are real and its eigenvectors are orthogonal, i.e. their pairwise inner products are zero (in other words, they are linearly independent). The vector $\mathbf{1} = \mathbf{e} \triangleq (1,1,\dots,1)^{\mathrm{T}} \in \mathbb{R}^{\mathrm{N}}$ is an eigenvector of the Laplacian matrix with the associated eigenvalue of 0.

According to the Min-Max theorem (a.k.a. Courant-Fischer-Weyl theorem or principle), as the Laplacian matrix is Hermitian, and the minimum values of the Rayleigh-Ritz quotients with respect to the Laplacian matrix are its eigenvalues, while the arguments which make the Rayleigh-Ritz quotients minimum are the eigenvectors. Formally, the Rayleigh-Ritz quotient with respect to a matrix $\mathbf{M}$ is the fraction below, where $\langle \cdot,\cdot \rangle$ denotes inner product:

$$R_M(x) = \frac{\langle x, Mx \rangle}{\langle x, x \rangle} = \frac{x^T M x}{x^T x} \quad (82)$$

$$R_L(x) = \frac{\langle x, Lx \rangle}{\langle x, x \rangle} = \frac{x^T L x}{x^T x} = \frac{\sum_{(i,j)\in L}(x(i)\text{-}x(j))^2}{\sum_{i\in N}(x(i))^2} \quad (83)$$

We follow a convention that sorts eigenvalues and their eigenvectors in an ascending order, i.e. $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$; therefore:

$$\lambda_1 = min\{R_L(x)|x \neq 0\} = \min_{x\neq 0}\frac{x^T L x}{x^T x} \quad (84)$$

$$v_1 = \{x|R_L(x) = \lambda_1\} = arg\min_{x\neq 0}\frac{x^T L x}{x^T x} \quad (85)$$

$$\lambda_n = max\{R_L(x)|x \neq 0\} = \max_{x\neq 0}\frac{x^T L x}{x^T x} \quad (86)$$

$$v_n = \{x|R_L(x) = \lambda_n\} = arg\max_{x\neq 0}\frac{x^T L x}{x^T x} \quad (87)$$

In other words, the value of Rayleigh-Ritz Quotient varies between the minimum (first) and maximum (last) eigenvalue of the matrix, and hence the name Min-Max theorem.

Another interesting interpretation of the Rayleigh-Ritz Quotient is based on the idea of vector functions as crisp or Fuzzy sets. This interpretation is mainly adopted from lecture notes of Daniel A. Spielman (Spielman, 2011) and (Spielman, 2007). Suppose $x \in \mathbb{R}^N$ is describing the membership values of a crisp set defining whether a street is blocked by construction work or not. For each street that is blocked we will have a 1 and for those not blocked we will have a 0 in the membership vector. As to the either-or nature of this membership function, this is a crisp set. We can also think of other sets with variable degrees of membership in the range $[0,1]$ defining, for instance, to what degree a street is polluted; this will be a Fuzzy set. The Rayleigh-Ritz Quotient then will be a descriptor of how relatively sparse the set in question is. This is because the nominator of the Rayleigh-Ritz Quotient would measure the size of the boundary (the edges from the set towards those not in the set) of the set in question as it only gets a non-zero value in the links between inside and outside. Everywhere else, the term $(x(i)\text{-}x(j))^2$ for a link (i,j) would be zero. The denominator also describes the size of the set in question as it only counts the non-zero elements. Therefore, the Rayleigh-Ritz Quotient can be seen as a descriptor of the relative sparsity of a cluster defined by a set and its vector valued membership function.

§ 5.12.2 Degree Centrality

The first measure of centrality that often comes to the minds of people when they look at a network, as an obvious measure of importance, is degree of nodes, i.e. how many neighbours a node has. On one hand, this measure is quite simplistic and tends to ignore the entirety of the network structure in that it does not capture anything beyond the immediate neighbours of a node. In this sense, it is an inherently local measure of centrality. On the other hand, we shall see that degree centrality, when normalized can be also seen as a simple probabilistic model, i.e. the stationary probability of a Markov Chain. To prepare for the next definition note that degree centrality can be seen as accounting for walks of length one-step, i.e. to immediate neighbours. Normalized Degree Centrality for a node is defined as a degree of the node in question divided by the sum of degrees all over the network. From elementary Graph Theory, we know that this sum equals the number of links (edges) multiplied by two. Formally:

$$c^d(n_i) = \frac{d_i}{\sum_i d_i} = \frac{d_i}{2|L|} \quad (88)$$

FIGURE 120 shows normalized degree centrality

§ 5.12.3 Eigenvector Centrality Metrics (Katz, Gould, Bonacich)

If we change the definition of importance from "how many nodes (streets) are connected to a node (street)" to a more subtle notion as: important nodes are those connected to important nodes, then we arrive at a definition that is in fact an eigenvector of the adjacency matrix, hence the name eigenvector centrality. This definition in fact covers a number of metrics originated within the fields of Psychometrics (Katz, 1953), Social Network Analysis (Bonacich, 1972), and Geography (Gould, 1967). In case of a social network, we can reformulate the definition as "an important person is a person who is connected to important people" or somebody who "knows anybody who is anybody". Similar notions exist such as the idiom "man is known by the company he keeps". In fact, Bonacich initially defined it with a different formulation that is also very interesting. His formulation was based on counting the number of walks from a node to any other node within the network -which in our case literally translates to the number of ways to get from one street to all other streets- while the walks of longer length are attenuated inversely as to their length. In this sense, eigenvector centrality can be seen as accounting for walks of maximum length possible in the network, that is, eigenvector centrality and degree centrality represent to poles of a spectrum. Bonacich, later in (Bonacich, 1987) introduced a parameter beta that allows attaining centrality measures anywhere desired on this spectrum. As Bonacich explains, this measure can be seen to be a sort of closeness centrality measure as it magnifies positions that have many short paths with high weights to all other locations. However, the advantage of eigenvector centrality is that it does not assume any optimal path, even our Easiest Path. Instead, it considers the randomness in choosing a path as it considers all possibilities. This spirit is shared among the rest of measures to be introduced by the end of this chapter, that they capture the stochastic nature of walking or wandering in networks. We here explain in detail how the two definitions of eigenvector centrality mean the same thing. The first formulation based on the recursive notion that says one is not necessarily important in a social network just because they have many friends, but one can be important for having important friends. Similarly, a street can be seen as important in a city network because of being connected to important streets. Therefore, a (weighted) sum of importance of neighbours or connections should be proportionate to the importance of a node. What follows is a slightly different explanation of the metric. Formally:

$$c^e(i) \propto \sum_{j\sim i} c^e(j) \tag{89}$$

$$c^e(i) = \eta \sum_{j\sim i} c^e(j) = \eta \sum_{j=0}^{n-1} a_{i,j}\, c^e(j) \tag{90}$$

In matrix form:

$$\boldsymbol{c}^e = \eta \boldsymbol{A}\boldsymbol{c}^e \tag{91}$$

This would be more interesting if we reformulate as below where $\lambda = 1/\eta$

$$\boldsymbol{A}\boldsymbol{c}^e = \lambda \boldsymbol{c}^e \tag{92}$$

That is clearly in the form of an eigensystem like $\boldsymbol{A}\boldsymbol{x} = \lambda \boldsymbol{x}$, hence the alias eigenvector centrality for the Power Index as was initially dubbed by Bonacich. He then generalized this definition by deeming that a node can be dependant to a certain degree on its neighbours, as if it has also some importance of its own. This is done by reformulating the same concept by inserting a parameter β for adjusting the effect of connections and a parameter α to account for what importance the node in question already has:

$$\boldsymbol{c}^e(i) = \sum_{j=0}^{n-1} (\alpha + \beta \boldsymbol{c}^e(j)) a_{i,j} \tag{93}$$

$$\boldsymbol{c}^e(i) = \alpha \sum_{j=0}^{n-1} a_{i,j} + \beta \sum_{j=0}^{n-1} a_{i,j}\, \boldsymbol{c}^e(j) \tag{94}$$

In matrix form:

$$\boldsymbol{c}^e = \alpha \boldsymbol{A}\boldsymbol{e} + \beta \boldsymbol{A}\boldsymbol{c}^e \tag{95}$$

$$\boldsymbol{c}^e \text{-} \beta \boldsymbol{A}\boldsymbol{c}^e = \alpha \boldsymbol{A}\boldsymbol{e} \tag{96}$$

$$\boxed{(\boldsymbol{I} - \beta \boldsymbol{A})\boldsymbol{c}^e = \alpha \boldsymbol{A}\boldsymbol{e}} \tag{97}$$

$$\boxed{\boldsymbol{c}^e = \alpha (\boldsymbol{I} - \beta \boldsymbol{A})^{-1} \boldsymbol{A}\boldsymbol{e}} \tag{98}$$

It is interesting to see that $\boldsymbol{Ae} = \boldsymbol{d}$, where $\boldsymbol{d}$ is a column vector containing node degrees. The parameter alpha turns out to be a normalization parameter, which can be set in such a way as to ensure that the 2-norm of the centrality vector is equal to one, i.e. $\|\boldsymbol{c}^e\| = \sum_i (\boldsymbol{c}^e(i))^2 = 1$. Now, let us see how this formulation can account for all walks from a node (street). Using the first equation in matrix form:

$$\boldsymbol{c}^e = \alpha\boldsymbol{Ae} + \beta\boldsymbol{Ac}^e = \alpha\boldsymbol{Ae} + \beta\boldsymbol{A}(\alpha\boldsymbol{Ae} + \beta\boldsymbol{Ac}^e) \tag{99}$$

$$\boldsymbol{c}^e = \alpha\boldsymbol{Ae} + \beta\boldsymbol{A}(\alpha\boldsymbol{Ae} + \beta\boldsymbol{Ac}^e) = \alpha\boldsymbol{Ae} + \alpha\beta\boldsymbol{A}^2\boldsymbol{e} + \beta^2\boldsymbol{A}^2\boldsymbol{c}^e \tag{100}$$

$$\boldsymbol{c}^e = \alpha\boldsymbol{Ae} + \alpha\beta\boldsymbol{A}^2\boldsymbol{e} + \beta^2\boldsymbol{A}^2(\alpha\boldsymbol{Ae} + \beta\boldsymbol{Ac}^e) \tag{101}$$

$$\boldsymbol{c}^e = \alpha\boldsymbol{Ae} + \alpha\beta\boldsymbol{A}^2\boldsymbol{e} + \alpha\beta^2\boldsymbol{A}^3\boldsymbol{e} + \beta^3\boldsymbol{A}^3\boldsymbol{c}^e \tag{102}$$

Therefore, it is clear and provable by induction that:

$$\boldsymbol{c}^e = \lim_{k\to\infty}(\alpha\boldsymbol{Ae} + \alpha\beta\boldsymbol{A}^2\boldsymbol{e} + \alpha\beta^2\boldsymbol{A}^3\boldsymbol{e} + \cdots + \alpha\beta^k\boldsymbol{A}^{k+1}\boldsymbol{e}) \tag{103}$$

$$\boldsymbol{c}^e = \alpha\sum_{k=0}^{\infty}\beta^k\boldsymbol{A}^{k+1}\boldsymbol{e} \tag{104}$$

As apparent in both formulations, α turns out to be only a scale factor for the centrality indices. It can be chosen so that the eigenvector is normalized or that its 2-norm equals one, depending on the context. This series converges to its limit provided the beta parameter is smaller than the reciprocal of the largest eigenvalue of the adjacency matrix. We shall shortly see a justification for this convergence; but first let us see an interesting connection with geographical accessibility. It is particularly of methodological interest in that it shows how similar the methods in geographical analysis and social network analysis can be. Peter Gould proposed his Index of Accessibility (Gould, 1967) much earlier than the work of Bonacich. He proposed to use the prominent eigenvector of the adjacency matrix associated with a road network as a measure of accessibility for very similar reasons.

To see literally, why this name makes sense, observe the last equation derived above that shows that the eigenvector centrality is proportional to row sums of powers of the adjacency matrix. It is a simple fact in Graph Theory the $(i,j)^{th}$ entries of an adjacency matrix raised to a power k refer to the number of paths of length k between the (i,j) pair of nodes. We adapt our justification from the elegant description of (Spizzirri, 2011); he asserts that the row sums of powers of adjacency matrices have been used even before the introduction of the two mentioned eigenvector centrality measures in analytic geography. The reasoning is that the row sums of an adjacency matrix raised to a power show how many paths of that length exist from a node to all other nodes, hence it literally describes the access of that node to all other locations or persons in the network. The following description, after (Spizzirri, 2011) justifies the convergence of the abovementioned series and provides more insight as to why eigenvectors and eigenvalues are important in assessing accessibility in a stochastic sense.

By virtue of orthogonality of eigenvectors of an $N \times N$ symmetric matrix, we can write any vector $\boldsymbol{x} \in \mathbb{R}^N$ as a linear combination of eigenvectors (For the sake of clarity of notations, we denote the number of nodes in the graph $|N|$ as n, as in our computational implementations):

$$\boldsymbol{x} = \alpha_1 \boldsymbol{v}_1 + \alpha_2 \boldsymbol{v}_2 + \cdots + \alpha_n \boldsymbol{v}_n \tag{105}$$

Multiplying both sides from left by the adjacency matrix:

$$\boldsymbol{A}\boldsymbol{x} = \alpha_1 \boldsymbol{A}\boldsymbol{v}_1 + \alpha_2 \boldsymbol{A}\boldsymbol{v}_2 + \cdots + \alpha_n \boldsymbol{A}\boldsymbol{v}_n \tag{106}$$

By virtue of the fact that vectors $\boldsymbol{v}_i$ are eigenvectors with corresponding eigenvalues as λ_i we can replace all $\boldsymbol{A}\boldsymbol{v}_i$ terms by $\lambda_i \boldsymbol{v}_i$ terms:

$$\boldsymbol{A}\boldsymbol{x} = \alpha_1 \lambda_1 \boldsymbol{v}_1 + \alpha_2 \lambda_2 \boldsymbol{v}_2 + \cdots + \alpha_n \lambda_n \boldsymbol{v}_n \tag{107}$$

$$\boldsymbol{A}^2\boldsymbol{x} = \alpha_1 \lambda_1 (\lambda_1 \boldsymbol{v}_1) + \alpha_2 \lambda_2 (\lambda_2 \boldsymbol{v}_2) + \cdots + \alpha_n \lambda_n (\lambda_n \boldsymbol{v}_n) \tag{108}$$

$$\boldsymbol{A}^k\boldsymbol{x} = \alpha_1 {\lambda_1}^k \boldsymbol{v}_1 + \alpha_2 {\lambda_2}^k \boldsymbol{v}_2 + \cdots + \alpha_n {\lambda_n}^k \boldsymbol{v}_n \tag{109}$$

In order to proceed with the reasoning we need to refer to the **Perron-Frobenius theorem**, which requires a few definitions: A matrix $\boldsymbol{M}$ is called non-negative if all of its entries are non-negative, i.e. $m_{i,j} \geq 0 \; \forall \; i,j$. A square non-negative matrix $\boldsymbol{A}$ is called primitive if there exists an integer $K > 0$, such that $\boldsymbol{A}^K$ is positive, i.e. $a_{i,j} > 0 \; \forall \; i,j$. The Perron-Frobenius theorem then asserts that for a primitive matrix there is a positive real eigenvalue r, which is called **spectral radius** or Perron-Frobenius eigenvalues, corresponding to which there is an eigenvector with all positive entries. This eigenvalue is strictly greater than other eigenvalues[a]. The integer K in our case is the maximum number of streets that can be considered the graph-theoretical distance between two streets (nodes) in a street network. If the graph is connected (i.e. every street is accessible from any street), its adjacency matrix when raised to a power larger than K is guaranteed to have all positive entries. This is because these entries

a Note that since we are studying undirected graphs at this point, their adjacency matrices are symmetric (which means they are self-adjoint or Hermitian in that they equal their conjugate transposed versions), and hence their eigenvalues are real and they are guaranteed to have linearly independent eigenvectors.

enumerate the number of paths of length K between nodes and we have ensured that K is big enough to guarantee the existence of such a path for the farthest possible pair of nodes. Two important conclusions arise from this theorem:

- We seek a range of positive values as a centrality index vector and hereby their existence as such is ascertained.
- Considering the fact that this eigenvalue is strictly greater than other eigenvalues, it is clear that it decides the limit of the abovementioned series. We can see this fact by dividing both sides of the equation by ${\lambda_n}^k$ (considering our convention for sorting eigenvalues in an ascending order, $\lambda_n = r$ would be the prominent or Perron-Frobenius eigenvalue):

$$\frac{A^k x}{{\lambda_n}^k} = \frac{\alpha_1 {\lambda_1}^k v_1}{{\lambda_n}^k} + \frac{\alpha_2 {\lambda_2}^k v_2}{{\lambda_n}^k} + \cdots + \alpha_n v_n \quad (110)$$

$$\lim_{k\to\infty} \frac{A^k x}{{\lambda_n}^k} = \lim_{k\to\infty} \sum_{i=1}^{n} \frac{\alpha_i {\lambda_i}^k v_i}{{\lambda_n}^k} = \alpha_n v_n \quad (111)$$

This is because the other terms will quickly approach zero:

$$\frac{\alpha_i {\lambda_i}^k v_i}{{\lambda_n}^k} = \alpha_i \frac{{\lambda_i}^k}{{\lambda_n}^k} v_i = \alpha_i \left(\frac{\lambda_i}{\lambda_n}\right)^k v_i \quad (112)$$

$$\lambda_i < \lambda_n \; \forall i \in N \xrightarrow{yields} \lim_{k\to\infty} \alpha_i \left(\frac{\lambda_i}{\lambda_n}\right)^k \boldsymbol{v}_i = 0 \quad (113)$$

A very closely related centrality measure that dates even before the two already mentioned was constructed by Leo Katz in 1953, for very similar reasons, but with a different name, as a measure of status in a social network (Katz, 1953). Katz centrality is a generalization of degree centrality in the sense that it considers status or influence as being dependant on the total number of ways one has access to people in a network through already existing connections. This is very much like the possibility of becoming introduced to someone in a professional network (such as LinkedIn) through those to whom one is already connected and their contacts and so forth. This formulation turns out to be almost identical to the Power Centrality as explained above. Formally, he defines the centrality of a node as the column sums of all powers of the adjacency matrix multiplied by an attenuation factor β (in a closely related formulation called alpha centrality, this parameter is dubbed α). The following description is adopted from (Borgatti, Stephen P., Everett, Martin G., 2006) and that of (Katz, 1953). Katz build his argument based on a generalization of the notion that the column sums of an adjacency matrix give the number of ‘choices’ available to an individual in a network (say to how many streets one can go from one), that is in fact the degree of that node.

Then he continues that the column sums of $\boldsymbol{A}^2$ give the number of two-step choices (say two streets away) and so forth. Hence, he figured that a matrix containing the 'weight' of all such choices $\boldsymbol{W}$ is of importance, provided the longer access routes have a lower impact on the outcome. This is guaranteed by multiplying the adjacency matrix with an attenuation factor dubbed β (also called alpha in some alternative notations).

$$w_{(i,j)} = \lim_{k\to\infty} \left(\beta a_{(i,j)} + \beta^2 a^2_{(i,j)} + + \beta^3 a^3_{(i,j)} + \cdots + \beta^k a^k_{(i,j)}\right) = \sum_{k=1}^{\infty} \beta^k a^k_{(i,j)} \quad (114)$$

$$\boldsymbol{c}^{Katz} = \sum_{j} w_{(i,j)} \rightarrow \boldsymbol{c}^{Katz}(i) = \sum_{j=0}^{|N|-1} \sum_{k=1}^{\infty} \beta^k a^k_{(i,j)} \quad (115)$$

$$\boldsymbol{c}^{Katz} = \boldsymbol{We} \quad (116)$$

$$\boldsymbol{c}^{Katz} = (\beta \boldsymbol{A} + \beta^2 \boldsymbol{A}^2 + \cdots + \beta^k \boldsymbol{A}^k)\boldsymbol{e} \quad (117)$$

$$(\boldsymbol{I}\text{-}\beta\boldsymbol{A})(\boldsymbol{c}^{Katz}) = (\boldsymbol{I}\text{-}\beta\boldsymbol{A})\left(\sum_{k=1}^{\infty} \beta^k \boldsymbol{A}^k\right)\boldsymbol{e} \quad (118)$$

$$(\boldsymbol{I}\text{-}\beta\boldsymbol{A})\boldsymbol{c}^{Katz} = \left[\left(\sum_{k=1}^{\infty} \beta^k \boldsymbol{A}^k\right) - \left(\sum_{k=2}^{\infty} \beta^k \boldsymbol{A}^k\right)\right]\boldsymbol{e} = (\beta\boldsymbol{A})\boldsymbol{e} \quad (119)$$

$$\boxed{(\boldsymbol{I} - \beta\boldsymbol{A})\boldsymbol{c}^{Katz} = \beta\boldsymbol{Ae}} \quad (120)$$

$$\boxed{\boldsymbol{c}^{Katz} = \beta((\boldsymbol{I} - \beta\boldsymbol{A})^{-1}\boldsymbol{A})\boldsymbol{e}} \quad (121)$$

Practically, we solve the system of linear equations to compute Katz centrality, because inverting a matrix is very costly. Note that this formulation of centrality is almost identical to that of [generalized] Power centrality (Bonacich, 1987). The condition for convergence of the power series is also that the parameter β is smaller than the reciprocal of the dominant eigenvalue, as shown before. The results of Katz centrality for our tests data set are the same as eigenvector centrality when normalized. However, the advantage of Katz centrality is its usability for directed graphs, (it was initially defined on a directed social network).

FIGURE 121 shows Katz centrality computed with alpha parameter set to 0.01

FIGURE 122 shows generalized (Bonacich) Eigenvector Centrality, when beta is zero; this is identical algebraically to degree centrality. Note the similarity to Katz Centrality when beta is small.

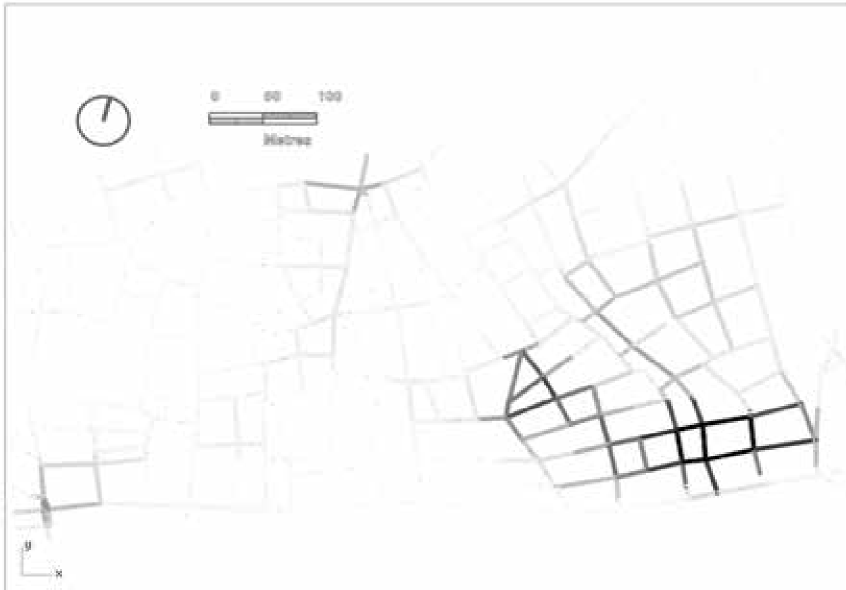


FIGURE 123 shows Fuzzy eigenvector centrality when "normalized beta" is set to 95% of its maximum, i.e. 1 over the spectral radius (0.2023). It turns out that the Fuzzy closeness matrix and the topological adjacency matrix have very similar normalized eigenvector centrality metrics as they both result in almost identical colouring. This extreme of eigenvector centrality takes the prevalent eigenvector of the connectivity matrix and results in a very sharp ranking. This is very similar to Google PageRank method for ranking important webpages as the pages that are connected to important pages. See the next topic.

In our implementation, we normalize beta parameter dub it eta η, to free user from a concern of convergence. First, we compute the spectral radius efficiently through Power Iteration algorithm and then algebraically compute the eigenvector centrality.

$$\eta = \beta.(\frac{1}{\rho}), \rho = spectral\ radius \tag{122}$$

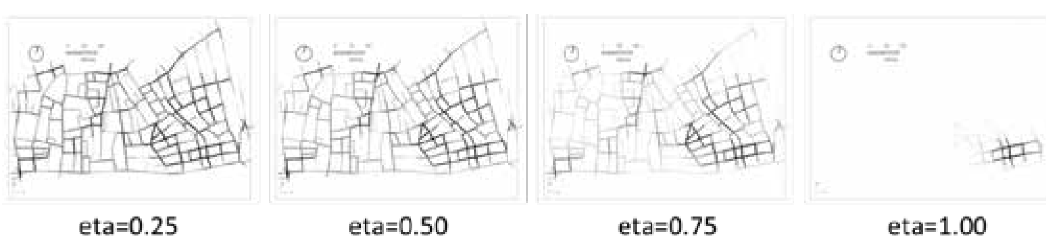


FIGURE 124 shows the effect of normalized beta (eta) parameter on Bonacich's eigenvector centrality

The fact that makes all these measures more interesting is that they can be computed on a weighted adjacency matrix as well. The notion of weight in Graph Theory literature is usually somewhat unclear regarding its physical meaning. In our work on spectral graph drawing and in here, we adopt a view that the entries of a weighted adjacency should represent admittance rather than impedance, or closeness rather than farness. This is a very important point as producing farness weights might not be straightforward as optimal path algorithms usually work with impedances. In our work, however, we have already created the means for making such matrices that is our Fuzzy nearness measure, mentioned already in this chapter. Note that the concept of degree as the number of nodes immediately connected to a node generalizes to the sum of weights of nodes connected to a node in case of weighted adjacency matrices. To shorten the discussion, consider that we can replace all adjacency matrices with a weighted adjacency matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ (a fuzzy graph) whose entries are defined as follows[65]:

$$a_{(i,j)} = \begin{cases} f\left(d_{(i,j)}\right) | f(x) = \dfrac{1}{1 + e^{-\mu\left(\frac{F}{2} - x\right)}}, d_{(i,j)} = \sum_{k \in \gamma_{(i,j)}} \zeta_k, & if\ d_{(i,j)} \neq \infty \\ 0, & if\ d_{(i,j)} = \infty \end{cases} \tag{123}$$

The definition of node degrees also changes to the row sums of the above matrix. The results of applying this weighted adjacency matrix are marked with the prefix Fuzzy wherever shown; but they generally correspond very well to the results achieved by the adjacency matrix. This shows the validity of this approach. If impedance (farness) weights used, the whole set of eigenvector centrality measures become meaningless and undefined.

One might think why we explained both Katz and eigenvector centrality metric of Bonacich and why we introduced fuzzy weights? They all produce very similar or identical distributions; but the point was to justify the same phenomenon from different perspectives and put them all together as one body of metrics.

§ 5.13 Probabilistic Models on Configuration Graphs

Centrality models are interesting in that they provide insight into relative importance of nodes in a graph; however, each centrality model produces a ranking that is only valid in the context of a particular definition of importance. Therefore, we cannot easily say which centrality metric is absolutely better than another is. If we adopt the view that the actual spatial behaviour of pedestrians and cyclists might not be as purposeful as assumed in a model such as betweenness centrality, we can see that this model does not consider any randomness in the mobility phenomenon. On the other hand, the eigenvector centrality models do not assume anything on the nature of the paths and their optimality; they consider all paths, irrespective of their optimality. We believe that the actual behaviour of pedestrians and cyclists might be somewhere in between: neither entirely random nor entirely purposeful or well planned. An interesting question is whether we can go model the probability of presence or passage of people using mathematical models. We have in fact done this already with our normalized model of betweenness centrality, as it describes the chance of having an Easiest Path passing through a street relative to the number of all possible Easiest Paths. That model can be interpreted as follows: if everybody knows "what is the best way to get from A to B" and that they all have a direct well-planned trip, what would be the chance of finding one of these determined people on a certain street. In fact, as we mentioned before, this model has been shown to be relevant in terms of its association with the location of retail businesses (which heavily depend on the probability of passage of pedestrians). From an entirely different perspective, that model is revealing something about the nature of the network and its 'potentials'; so it suits planning purposes. However, if we were to model such probabilities adopting an alternative view which considers a degree of randomness in the choices people make in going from somewhere to somewhere else, we should go for different category of models called stochastic models. The models we introduce as such are all based on **Markov Chains** (alias **Random Walk** or **Google PageRank**).

§ 5.13.1 A Markov Chain Model/ Random Walks on Streets

A Markov Chain (attributed to *Andrei Andreyevich Markov* 1856-1922) is a versatile mathematical model used to model stochastic phenomena. It has been used for studying urban networks and their properties previously, see for instance: (Volchenkov, D., and Ph Blanchard., 2007), (Volchenkov, D, Blanchard, P, 2007) (Blanchard, Philippe, and Dimitri Volchenkov, 2008), (Jiang, 2009), (Fidler, Dror & Hanna, Sean, 2015) .

A Markov Chain is not aimed at predicting individual stochastic moves of agents or the transient states of a system; instead, it is used to study the long term 'steady state' of a stochastic system. In our case, if we are interested in the expected percentage of pedestrians or cyclists in the streets of a city or neighbourhood, we can represent the states of the street network as a system, formally, as a 'row vector' $\boldsymbol{x} \in \mathbb{R}^n$, in which $\boldsymbol{x}(i)$ would be the probability of finding somebody on the i^{th} street. In other words, the vector $\boldsymbol{x}$ is a probability distribution, which entails the following:

$$\forall i \in N: \boldsymbol{x}(i) \in [0,1] \text{ and } \|\boldsymbol{x}\|_1 = \sum_{i=0}^{n-1} \boldsymbol{x}(i) = 1 \quad (124)$$

The possible values of $\boldsymbol{x}$ are referred to as the state space of the system understudy. The base of a Markov Chain is the assumption that the system understudy does not have a memory of its past, i.e. its current state only depends on the previous state. Formally, if we consider a discrete time random walk, composed of steps, across which the state of the system transits from the state $\boldsymbol{x}^t$, i.e. the state of the system at the discrete time step t to another state at the next moment in time, which is denoted as $\boldsymbol{x}^{t+1}$. In order to construct a Markov Chain model, we need to be able to model the probability of transiting from one state to a consecutive state. The collection of transition probabilities is conventionally represented as a matrix $\boldsymbol{P} \in \mathbb{R}^{n \times n}$, called the Transition Probability Matrix. The memoryless-ness property of the Markov Chain (alias Markovian Property) is then formally described in terms of conditional probabilities as below (in this context, we use superscripts in parentheses for denoting discrete time, not power):

$$\Pr\left(\boldsymbol{x}^{(t+1)} = \boldsymbol{j} \middle| \boldsymbol{x}^{(0)} = \boldsymbol{k}_0, \boldsymbol{x}^{(1)} = \boldsymbol{k}_1, \ldots, \boldsymbol{x}^{(t)} = \boldsymbol{i}\right) = \Pr\left(\boldsymbol{x}^{(t+1)} = \boldsymbol{j} \middle| \boldsymbol{x}^{(t)} = \boldsymbol{i}\right) \quad (125)$$

In other words, the conditional probability of being at a state only depends on the previous state, and not all other previous states. In our case, the states of the system are literally represented by the nodes in the graph, i.e. the state of being at the i^{th} street is represented with following vector:

$$\boldsymbol{e}_i \in \mathbb{R}^n | \boldsymbol{e}_i(k) = \begin{cases} 1, k = i \\ 0, otherwise \end{cases}, \forall\, k \in [1, n] \quad (126)$$

However, for the sake of simplicity of our notations, we denote such states literally with the index of the non-zero entry in bold lowercase letters[a], e.g. the above state is represented as $\boldsymbol{i} = \boldsymbol{e}_i$ ($\boldsymbol{e}$ defined as a row vector). Observe that if a state vector as such has more than one non-zero entry then they should all represent probabilities and add up to 1, this is because the system at a time can only have one state, that is the position of a 'random walker' who cannot be at more than one place at a time. Then, the Transition Probability matrix $\boldsymbol{P} \in \mathbb{R}^{n \times n}$, is defined as below:

$$p_{(i,j)} = \Pr\left(\boldsymbol{x}^{(t+1)} = \boldsymbol{j} \middle| \boldsymbol{x}^{(t)} = \boldsymbol{i}\right) \quad (127)$$

The above equation in our discrete spatial network model translates to the following: the conditional probability that a random walker goes to the j^{th} street in his next move

a This notation is in line with the convention notation for Cartesian unit vectors that form a standard basis for Euclidean space of $\mathbb{R}^3$: $\boldsymbol{i} = [1,0,0]^T, \boldsymbol{j} = [0,1,0]^T, \boldsymbol{k} = [0,0,1]^T$.

with the condition that he is currently at the i^{th} street is denoted as $p_{(i,j)}$. Note that the above definition derived from the memoryless-ness property implies that the transition probabilities are independent of transition time. We shall shortly define them in terms of spatial properties we have already measured. In the meantime, we need to explain the notion of a Random Walk, which is methaphorically described as the mobility behaviour of a 'walking drunkard' whose choices for going to any direction are random, and supposedly following a probability distribution. If we consider a 'Lazy Random Walk' action, in which the walker in question might choose to linger at a street with a certain probability, then:

- The diagonal entries $p_{(i,i)}$ will be non-zero probabilities of **staying** at a node (street); and
- The off-diagonal entries $p_{(i,j)}$ will be describing the probability of **leaving** the street corresponding to the row index and transiting to the street corresponding to the column index.

He cannot fly to just any street in the network; therefore, his choices would be limited to the neighbouring streets. In other words, his movement trajectories are necessarily bound to the network. As said above, the chances of going to the next street are subject to a probability distribution. This entails that the row sums of the matrix $\boldsymbol{P}$ should be equal to one. A matrix as such is called a stochastic matrix. Before we go further with constructing such a matrix, we remind the use of a Markov Chain model: The purpose of constructing a transition probability matrix (that is time invariant) is to compute the stationary probability distribution of random walkers (and cyclists!) by it. We shall see how the stationary probability distribution relates to this matrix; but first, we focus on the matrix itself.

§ 5.13.2 Four Different Random Walk models

Without loss of generality and in the absence of spatial attributes, i.e. ceteris paribus, we can assume that the probability of hopping to a neighbouring street has to do with the ease of transition from i^{th} street to j^{th} street. Having computed the Fuzzy closeness values according to the conditions set for the Easiest Path algorithm, we can assume that Fuzzy closeness values represent this ease in an all-inclusive 'absolute' manner. For a probability distribution, we can define 'relative' ease of hopping to a neighbour as being proportionate to the relative ease of the transition in question over all possible transitions. We denote the Fuzzy closeness of the i^{th} street to j^{th} street as $c_{(i,j)}$, which is obtained by applying the Fuzzy closeness function defined before to the Easiest Path distance of j^{th} street to i^{th} street, as computed before. That is to say, we will use the entries of our 'Fuzzy weighted adjacency matrix' whose node 'degrees', consistent with their crisp counterparts, are defined as row sums of the weighted adjacency matrix, meaning:

$$d_i = \sum_{j \sim i} c_{(i,j)} \tag{128}$$

Therefore, in order to relativize closeness values, we divide them by their local sums (weighted degrees as defined above):

$$p_{(i,j)} \propto \frac{c_{(i,j)}}{\sum_{j \sim i} c_{(i,j)}} = \frac{c_{(i,j)}}{d_i} \tag{129}$$

$$p_{(i,j)} = \xi \frac{c_{(i,j)}}{d_i} \tag{130}$$

To change the proportionality relation to an equality relation we use ξ (Greek letter Xi) as a normalizing parameter. In case we did not consider this a lazy walk, it would be easy to ensure that matrix $\boldsymbol{P}$ is stochastic, i.e. its row sums represent probability distributions and hence add up to one, the normalizing parameter should be chosen as one. Because:

$$\sum_{j \sim i} \xi \frac{c_{(i,j)}}{d_i} = \frac{\xi}{d_i} \sum_{j \sim i} c_{(i,j)} = \frac{\xi}{d_i} d_i = 1 \xrightarrow{yields} \xi = 1 \tag{131}$$

Therefore, for a non-lazy Random Walk, that we cardinally call model Alef, and denote by Hebrew superscript $\aleph$:

$$p^{\aleph}_{(i,j)} = \frac{c_{(i,j)}}{d_i} \tag{132}$$

This equation in matrix form can be rewritten as below, by denoting $\boldsymbol{C} = [c_{(i,j)}]_{n \times n}$ and denoting $\boldsymbol{D}$ as a diagonal matrix whose diagonal entries equal node degrees, i.e. $d_{(i,i)} = d_i$ or generally written as $d_{(i,j)} = \delta_{(i,j)} d_i$, where $\delta_{(i,j)} = \{1, iff\ i = j\ \}$ is Kronecker delta and $\boldsymbol{P} = [p_{(i,j)}]_{n \times n}$:

$$\boxed{\boldsymbol{P}^{\aleph} = \boldsymbol{D}^{-1}\boldsymbol{C}} \tag{133}$$

Note that this definition is a generalized version of the matrix $\boldsymbol{D}^{-1}\boldsymbol{A}$, which is often introduced as the default Random Walk transition matrix.

This equation holds because if we expand the matrix multiplication:

$$p^{\aleph}_{(i,j)} = \sum_{k=1}^{n} [\boldsymbol{D}^{-1}]_{(i,k)} c_{(k,j)} = \sum_{k=1}^{n} \frac{\delta_{(i,k)}}{d_i} c_{(k,j)} = \frac{c_{(i,j)}}{d_i} \tag{134}$$

The Fuzzy closeness function returns 1 as its limit for distance values equal to zero, which is reasonable. Let us see why:

$$c_{(i,j)} = \frac{1}{1 + e^{\mu(x_{(i,j)} - \frac{F}{2})}} | \mu = \frac{2}{F} \ln\left(\frac{1}{\varepsilon} - 1\right) \xrightarrow{x_{(i,i)}=0} c_{(i,i)} = \frac{1}{1 + e^{\mu(0 - \frac{F}{2})}} \tag{135}$$

$$c_{(i,i)} = \frac{1}{1 + e^{\mu(0 - \frac{F}{2})}} = \lim_{\varepsilon \to 0} \frac{1}{1 + e^{(\frac{2}{F} \ln(\frac{1}{\varepsilon} - 1))(-\frac{F}{2})}} \tag{136}$$

$$c_{(i,i)} = \lim_{\varepsilon \to 0} \frac{1}{1 + e^{-\ln(\frac{1}{\varepsilon} - 1)}} = \lim_{\varepsilon \to 0} \frac{1}{1 + \frac{1}{e^{\ln(\frac{1}{\varepsilon} - 1)}}} = \lim_{\varepsilon \to 0} \frac{1}{1 + \frac{1}{\left(\frac{1}{\varepsilon} - 1\right)}} \tag{137}$$

$$c_{(i,i)} = \lim_{\varepsilon \to 0} \frac{1}{1 + \frac{1}{\left(\frac{1}{\varepsilon} - 1\right)}} = \lim_{\varepsilon \to 0} \frac{1}{1 + \frac{\varepsilon}{1 - \varepsilon}} = \lim_{\varepsilon \to 0} \frac{1}{\frac{1 - \varepsilon + \varepsilon}{1 - \varepsilon}} = \lim_{\varepsilon \to 0} (1 - \varepsilon) = 1 \tag{138}$$

This reads as 'every place is 'absolutely' close to itself' which is very reasonable of course, showing the consistency of our Fuzzy framework, but it also means if we formulate the probability of staying as a function of $c_{(i,j)}$ then by definition it would always be bigger than any probability of leaving. This is not desirable; therefore, we have to choose alternative probability values for $p_{(i,i)}$. We can either put them all to zero, as if the random walker never stops walking or we choose to model the probability of staying as a function of an external variable defining the attractiveness of a certain street. This could be for instance based on another centrality model such as closeness, which can essentially indicate the potential of a place for being a common destination that is likely to attract 'movement to' itself. We opt to first show a simpler formulation or a Lazy Random Walk, whose staying probabilities are equal to a constant. Such models are widely used to study such phenomena as diffusion in discrete spaces.

Having noted the ideal meaning of $c_{(i,j)}$, for the sake of simplicity of our notations and convenience in deriving our models hereafter we submit to the convention of $c_{(i,j)}$=0.

We first show a different equality based on transition probabilities of model א (Alef), in order to construct our alternative Random Walk models called and marked by ב (Bet), ג (Gimel), and ד (Dalet). Formally, in case it is NOT a lazy walk (a notion adopted from Daniel Spielman's lecture notes[a]):

a http://www.cs.yale.edu/homes/spielman/561/2009/lect08-09.pdf

$$p^{(t+1)}(j) = \sum_{i\sim j} p^{\aleph}_{(i,j)} p^{(t)}(i) = \sum_{i\sim j} \frac{c_{(i,j)}}{d_i} p^{(t)}(i) \tag{139}$$

This equation reads as 'the probability of finding a random walker being at a node j at time $t+1$ equals the sum of probabilities of finding a random walker at a neighbouring node (street) and the probability that this potential random walker chooses to transit to the node (street) understudy '.

In case of a Lazy Random Walk, specifically, if our random walker has a constant probability of staying at each node denoted as $\psi \in [0,1)$ then the probability of leaving that node to a neighbour should equal to 1- ψ. This model turns out to be a PageRank-like probabilistic model. That is:

$$p^{(t+1)}(j) = \psi p^{(t)}(j) + (1-\psi) \sum_{i\sim j} \frac{c_{(i,j)}}{d_i} p^{(t)}(i) \tag{140}$$

This equation reads as 'the probability of finding a random walker being at a node j at time $t+1$ equals the total sum of two probabilities:

- the probability that the random walker has been at node j before, i.e. ψ;
- or, if the walker has come from somewhere else, they must have come from one of the neighbouring nodes, that is the sum of probabilities of finding a random walker at a neighbouring node (street) AND the probability that this potential random walker chooses to transit to the node (street) understudy '.

We denote the probabilities of being at nodes of the network in vector form as $\boldsymbol{p}^{(t)}$, i.e. a column vector containing the probabilities of being at each node at discrete time t; hence the above equation can be written in matrix form as follows:

$$\boldsymbol{p}^{(t+1)} = (\psi \boldsymbol{I} + (1-\psi)\boldsymbol{D}^{-1}\boldsymbol{C})\ \boldsymbol{p}^{(t)} \tag{141}$$

Therefore, we define the Markovian transition probability matrix of model ב as follows:

$$\boxed{\boldsymbol{P}^{ב} \stackrel{\text{def}}{=} \psi \boldsymbol{I} + (1-\psi)\boldsymbol{D}^{-1}\boldsymbol{C}} \tag{142}$$

To verify that this is indeed a stochastic matrix, it helps to use an alternative notation as below:

$$p^{ב}_{(i,j)} = \begin{cases} \psi & , if\ j = i \\ (1-\psi)\left(\frac{c_{(i,j)}}{d_i}\right) & , if\ j \neq i \end{cases} \quad (143)$$

Now we need to verify that the row sums of a matrix $\boldsymbol{P}^{ב}$ as such equal 1, i.e.:

$$\sum_{j\sim i}(1-\psi)\left(\frac{c_{(i,j)}}{d_i}\right) == (1-\psi) \quad (144)$$

$$\left(\frac{1-\psi}{d_i}\right)\sum_{j\sim i} c_{(i,j)} = (1-\psi) \quad (145)$$

Therefore the transition probabilities are verified.

Note that for $\psi = \frac{1}{2}$ our generalized model turns out to be in the form of the widely known Lazy Random Walk model: $\boldsymbol{P} = 0.5(\boldsymbol{I} + \boldsymbol{D}^{-1}\boldsymbol{A})$.

Now we can define a wider family of Lazy Random Walks that can be biased. Let us say we receive from an external source some normalized values indicating the relative attractiveness of the streets in our network for a potential random walker (who is not necessarily drunk!). By these external values, we represent 'relative attractiveness' or 'probability of staying' values or show them as a stochastic vector denoted as $\boldsymbol{\psi}$ such that:

$$\boldsymbol{\psi} = \{\psi(i) \in [0,1] | \sum_{i} \psi(i) = 1\} \quad (146)$$

Then for each state (street), we must ensure the sum of probabilities is one, i.e. the probability of staying at the i^{th} street (denoted as p^s_i) plus the probability of leaving it (denoted as p^l_i) should add up to one or that the transition probability matrix needs to be a row stochastic matrix. This is because the random walker cannot vanish from that street!

Using the same method we used for models $א\ and\ ב$, we define the probability of being at a node as a sum of probabilities:

$$p^{(t+1)}(j) = \psi(j)p^{(t)}(j) + \sum_{i\sim j}(1-\psi(j))\frac{c_{(i,j)}}{d_i}p^{(t)}(i) \quad (147)$$

This equation can be interpreted as 'the probability of finding a random walker being at a node *j* at time $t+1$ equals the total sum of two probabilities:

- the probability that the random walker has been at node *j* before, i.e. $\psi(j)$ meaning because of its attractiveness they have stayed for a moment;
- or, if the walker has come from somewhere else, they must have come from one of the neighbouring nodes, that is the sum of probabilities of finding a random walker at a neighbouring node (street) AND the probability that this potential random walker chooses to transit to the node (street) understudy , only according to the ease of transition'.

To verify that this is indeed a stochastic matrix, it helps to use an alternative notation as below:

$$\mathrm{p}^{\gimel}_{(i,j)} = \begin{cases} \psi(i) & ,if\ j = i \\ (1-\psi(i))\left(\frac{c_{(i,j)}}{d_i}\right) & ,if\ j \neq i \end{cases} \tag{148}$$

Now we need to verify that the row sums of a matrix $\boldsymbol{P}^{\gimel}$ as such equal 1, i.e.:

$$\sum_{j\sim i}(1-\psi(i))\left(\frac{c_{(i,j)}}{d_i}\right) == (1-\psi(i)) \tag{149}$$

$$\left(\frac{1-\psi(i)}{d_i}\right)\sum_{j\sim i} c_{(i,j)} = (1-\psi(i)) \tag{150}$$

Therefore, the transition probabilities are verified.

This equation can be rewritten in matrix form as below, defining a diagonal matrix $\boldsymbol{\Psi}$, whose diagonal entries are the relative attractiveness of the nodes for being at; that is $[\boldsymbol{\Psi}]_{(i,j)} = \delta_{(i,j)}\psi(i)$, again by using Kronecker delta:

$$\boldsymbol{p}^{(t+1)} = (\boldsymbol{\Psi} + (\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{D}^{-1}\boldsymbol{C})\ \boldsymbol{p}^{(t)} \tag{151}$$

Out of this formulation, we arrive at our next alternative for a transition probability matrix, hence the Markov matrix of Random Walk ג (Gimel):

$$\boxed{\boldsymbol{P}^{\gimel} \stackrel{\text{def}}{=} \boldsymbol{\Psi} + (\boldsymbol{I}-\boldsymbol{\Psi})\boldsymbol{D}^{-1}\boldsymbol{C}} \tag{152}$$

Alternatively, if we believe that the attractiveness values can also encourage a random surfer to visit a node, we can also formulate the transition probabilities as to the relative appeal of neighbouring streets using a weighted product model of the total aptitude of each option available to the random walker. Suppose the walker in question, currently at node i has a number of nodes j to go to, each of which has a relative 'appeal' in terms of the attractiveness values received, i.e. $\psi(j)$, while going to each of these nodes has the relative 'ease' of $c_{(i,j)}$: then using a simple Weighted Product Model, the aptitude of each option is defined as $c_{(i,j)}\psi(j)$. Then we take relative aptitude of each option as proportionate to the probability of that option being chosen, that is:

$$\Pr(j|i) \propto \frac{c_{(i,j)}\psi(j)}{\sum_{j\sim i} c_{(i,j)}\psi(j)} \quad (153)$$

To convert the proportionality relation to an equation we account for the total probability of leaving node i for node j, which should be equal to one minus the probability of staying at node i, i.e. $\psi(i)$. This is to say:

$$\Pr(j|i) = (1 - \psi(i))\frac{c_{(i,j)}\psi(j)}{\sum_{j\sim i} c_{(i,j)}\psi(j)} \quad (154)$$

We choose to denote the sum of attractiveness values weighted with their closeness as destinations as an attribute of a node i, and call it $\chi(i)$ [Greek letter Chi] that is:

$$\chi(i) = \sum_{j\sim i} c_{(i,j)}\psi(j) \quad (155)$$

Therefore, we define transition probabilities of our Random Walk model ד (Dalet) as below:

$$p^{ד}_{(i,j)} = \begin{cases} \psi(i) & , if\ j = i \\ (1-\psi(i))\dfrac{c_{(i,j)}\psi(j)}{\chi(i)} & , if\ j \neq i \end{cases} \quad (156)$$

Now we need to verify that the row sums of a matrix $\boldsymbol{P}^{ד}$ as such equal 1, i.e.:

$$\sum_{j\sim i}(1-\psi(i))\frac{c_{(i,j)}\psi(j)}{\chi(i)} == (1-\psi(i)) \quad (157)$$

$$\left(\frac{1-\psi(i)}{\chi(i)}\right)\sum_{j\sim i}c_{(i,j)}\psi(j) = (1-\psi(i)) \quad (158)$$

Therefore, the transition probabilities are verified.

In order to rewrite the transition probabilities in matrix form we introduce a diagonal matrix $\boldsymbol{X}$ [uppercase Greek Chi] such that:

$$[\boldsymbol{X}]_{(i,j)} \stackrel{\text{def}}{=} \chi(i)\delta_{(i,j)} \quad (159)$$

When written in matrix form:

$$\boldsymbol{p}^{(t+1)} = (\boldsymbol{\Psi} + (\boldsymbol{I}\text{-}\boldsymbol{\Psi})\mathbf{X}^{-1}\boldsymbol{C\Psi})\ \boldsymbol{p}^{(t)} \quad (160)$$

This way we can extract the Markov matrix of the model ד (Dalet) as below:

$$\boxed{\boldsymbol{P}^{ד} \stackrel{\text{def}}{=} \boldsymbol{\Psi} + (\boldsymbol{I}-\boldsymbol{\Psi})\mathbf{X}^{-1}\boldsymbol{C\Psi}} \quad (161)$$

We hereby verify our matrix form notation of the model ד as the most general case of our four models by finding its (i,j) entries by expanding the matrix multiplications. To proceed smoothly, we first define an auxiliary matrix $\boldsymbol{M} = (\boldsymbol{I}\text{-}\boldsymbol{\Psi})\mathbf{X}^{-1}\boldsymbol{C}$ and find its (i,j) entries as below[a]:

$$[\boldsymbol{M}]_{(i,j)} = \sum_{k=1}^{n}[(\boldsymbol{I}\text{-}\boldsymbol{\Psi})\mathbf{X}^{-1}]_{(i,k)}\,[\boldsymbol{C}]_{(k,j)} \quad (162)$$

$$[\boldsymbol{M}]_{(i,j)} = \sum_{k=1}^{n}\left(\frac{1-\psi(i)}{\chi(i)}\delta_{(i,k)}\delta_{(i,k)}\right)c_{(k,j)} \quad (163)$$

$$\delta_{(i,k)} = \begin{cases}1 & ,if\ i=k\\ 0, otherwise\end{cases} \xrightarrow{yields} [\boldsymbol{M}]_{(i,j)} = \frac{1-\psi(i)}{\chi(i)}\delta_{(i,i)}\delta_{(i,i)}c_{(i,j)} = \frac{1-\psi(i)}{\chi(i)}c_{(i,j)} \quad (164)$$

a An expansion of a matrix multiplication between two $n\times n$ matrices A and B is: $C = AB \xrightarrow{yields} [C]_{(i,j)} = \sum_{k=1}^{n}[A]_{(i,k)}\,[B]_{(k,j)}$

Now we are prepared to verify the matrix formulation of our last model by replacing $\boldsymbol{M}$ in its definition, i.e. $\boldsymbol{P}^{ד} = \boldsymbol{\Psi} + \boldsymbol{M}\boldsymbol{\Psi}$:

$$[\boldsymbol{P}^{ד}]_{(i,j)} = [\boldsymbol{\Psi}]_{(i,j)} + [\boldsymbol{M}\boldsymbol{\Psi}]_{(i,j)} = \boldsymbol{\psi}(i)\delta_{(i,j)} + \sum_{k=1}^{n}[\boldsymbol{M}]_{(i,k)}\,[\boldsymbol{\Psi}]_{(k,j)} \tag{165}$$

$$[\boldsymbol{P}^{ד}]_{(i,j)} = \boldsymbol{\psi}(i)\delta_{(i,j)} + \sum_{k=1}^{n}\frac{1-\boldsymbol{\psi}(i)}{\chi(i)}c_{(i,k)}\,\boldsymbol{\psi}(k)\delta_{(k,j)} \tag{166}$$

$$[\boldsymbol{P}^{ד}]_{(i,j)} = \boldsymbol{\psi}(i)\delta_{(i,j)} + \frac{1-\psi(i)}{\chi(i)}c_{(i,j)}\boldsymbol{\psi}(j)\delta_{(j,j)} = \begin{cases}\boldsymbol{\psi}(i) & ,if\ j=i\\ \dfrac{1-\psi(i)}{\chi(i)}c_{(i,j)}\boldsymbol{\psi}(j), & if\ j\neq i\end{cases} \tag{167}$$

Thus, our matrix notation of model ד is verified. Other three models can be verified in a similar way.

Our last Random Walk model simulates a truly “Biased Random Walk” on our Configuration Graph; that is, unlike most Random Walk models the probabilities of going from a node to a node to its neighbours are not evenly distributed. This means and that the user of the models can choose to bias the model with any centrality model or even actual data pertained to the attractively of streets. In other words, the last model actually gives rise to a family of combinatory models such as Random Walk biased with a local Closeness Centrality model and alike.

Now that we have constructed four alternative transition probability matrices, let us see how we can study the statistical properties of the system using these matrices. We therefore focus our attention on the stationary probability distributions of random walkers.

§ 5.13.3 Stationary Distributions of Undirected Graphs

We can write the transition probability equations in matrix form as below:

$$\Pr\left(\boldsymbol{x}^{(t+1)}\middle|\boldsymbol{x}^{(t)}\right) = \boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)}\boldsymbol{P} \tag{168}$$

This is because:

$$x^{(t+1)}(j) = p_{(j,j)}x^{(t)}(j) + \sum_{i\sim j} p_{(i,j)}x^{(t)}(i) = \sum_{i=1}^{n} [P]_{(i,j)}\, x^{(t)}(i) \tag{169}$$

Note that instead of writing conditional probabilities we denote them from here on as the state vectors themselves; this is because these state vectors are row stochastic vectors that actually represent the probabilities of all possible states of the system. In our case, these states literally correspond to being at each indexed street that is also a node in our configuration graph models. We can also write the probability distribution at the time $t + 1$ as a function of the transition probability matrix and its previous state t; that is:

$$x^{(t)} = x^{(t\text{-}1)}P \tag{170}$$

Therefore, by replacing this in the previous matrix equation and by virtue of induction (or using Chapman-Kolmogorov Equations[66]):

$$x^{(t)} = x^{(t\text{-}2)}P^2 = x^{(t\text{-}3)}P^3 = \cdots = x^{(t\text{-}k)}P^k \tag{171}$$

This implies that:

$$x^{(t)} = x^{(0)}P^t \tag{172}$$

Well, the interesting point about a Markov Chain model is that if the model has certain properties, then this distribution converges to a limit distribution, and that we can predict the long-term steady-state probability distribution of states by using only the transition probability matrix, i.e. that probability turns out to be even independent of the initial state. In other words, we will not predict the exact individual transitions of states or moves of a random walker, but we will find the likelihood of finding a random walker on any street if he has been walking for a very long time. Alternatively, that is in an alternative interpretation, we find that 'the expected number of times that a random walker would visit a certain node' is proportional to the steady-state distribution. Of course, the random walker is a metaphor to get an intuition about the system but the fact is that such models have been shown to be good predictors of socio-economic distributions and human movement within cities, even better than spatial centrality models such as Space Syntax. For example see (Jiang, 2009) and (Wei, Xuebin, and Xiaobai A. Yao., 2014).
In the remaining part of this section, we first mention why and under what conditions

such a model would converge to a steady state and then show that the steady state is an eigenvector of the transition probability matrix. We will also show that this model is a spatial version of **Google** PageRank algorithm (Page, Lawrence, Sergey Brin, Rajeev Motwani, and Terry Winograd., 1999) used for searching web pages of the world-wide-web (WWW) for the most important webpages. We first describe the conditions under which a random walk model is guaranteed to converge to a stationary distribution and then we present an analytic solution for finding the stationary distributions.

Ergodicity is the key term for describing the conditions under which the state probability distribution of a Markov Chain is guaranteed to converge to a steady state distribution. The term ergodic has stemmed from Greek terms " $\varepsilon\rho\gamma o\nu$ (ergon): work" and " $o\delta o\varsigma$ (odos): way" and it, in a manner of speaking, implies that there is a way to get from any state to any other state without falling into loops (in the context of Markov Chains). We hereby review the conditions for ergodicity of a Markov Chain. A Markov chain is:

- *Irreducible*[67], if it is possible to get from any state to any other state in the state space with a positive probability or in other words $\exists\, t | \forall i, j: \boldsymbol{P}^{t}_{(i,j)} > 0$; and,
- *Aperiodic*[68], if the "period":= greatest common divisor of expected number of iterations for a possible return to a state, i.e. the cycle lengths, is 1 for all states, in other words there should not be a regularity or period in return times to states; formally $\forall\, i : \gcd\{t: \boldsymbol{P}^{t}_{(i,j)} > 0\} = 1$; and,
- *Positive Time Recurrent*[69], if the expected return time to a state must be finite, that is the number of iterations for the random walk process to expect a return to the same state as it started the walk must be finite.

More precisely, a Markov Chain is said to be ergodic if all of its states are ergodic, i.e. they are Aperiodic and Positive Time Recurrent (which necessities irreducibility as well). Since we are working with connected undirected simple graphs, the conditions for ergodicity are all met. In our case, this corresponds to the connectedness of the configuration graph corresponding to the spatial network, i.e. in the model it should be possible to get from any street to any other. Our Markov Chain model is said to have another property on top of this and that is it has to be *Regular*, i.e., some power of its transition probability matrix is guaranteed to have only positive elements. It can be shown that if a Markov Chain is irreducible and aperiodic it is a *Regular* chain. The importance of this condition, for a Markov Chain with finite state space is the applicability of Perron-Frobenius theorem; and thus the guarantee that it converges to a stationary distribution, which can be found as an eigenvector of the transition probability matrix associated with the dominant eigenvalue of 1 (Rihard Weber, James Norris, Grimmett, Stirzaker, Ross, Aldous, Fill, Grinstead and Snell, 2011).

The steady state (alias stationary [probability] distribution) is conventionally denoted as $\boldsymbol{\pi}$, i.e. a row vector representing states, which will not change when multiplied with the transition probability matrix anymore. It can be shown that such a distribution exists, i.e. the system eventually converges to a stationary probability distribution if it is ergodic and regular:

$$\boldsymbol{\pi} = \lim_{t\to\infty} \boldsymbol{x}^{(t)} \tag{173}$$

$$\boldsymbol{\pi} = \lim_{t\to\infty} \boldsymbol{x}^{(0)}\boldsymbol{P}^{t} \tag{174}$$

Note the following is not a proof; it is only another explanation of the last equation to make it more understandable. The actual proof of this equation can be achieved through the ergodic theorem, which guarantees the existence of a stationary probability distribution for an ergodic Markov Chain as explained before. The previous equation also implies the followings:

$$\boldsymbol{\pi} = \lim_{t\to\infty} \boldsymbol{x}^{(0)}\boldsymbol{P}^{t} \tag{175}$$

$$\boldsymbol{\pi} = \lim_{t\to\infty} \boldsymbol{x}^{(t-1)}\boldsymbol{P} \tag{176}$$

$$\boldsymbol{\pi} = \left(\lim_{t\to\infty} \boldsymbol{x}^{(t-1)}\right)\boldsymbol{P} \tag{177}$$

Back to our definition of $\boldsymbol{\pi}$, obviously the above limit is the same as this:

$$\boldsymbol{\pi} = \lim_{t\to\infty} \boldsymbol{x}^{(t)} = \lim_{t\to\infty} \boldsymbol{x}^{(t-1)} \tag{178}$$

Therefore, we can rewrite the above equation as:

$$\boldsymbol{\pi} = \boldsymbol{\pi}\boldsymbol{P} \tag{179}$$

We propose that P^{t} must be a $\mathrm{n}\times\mathrm{n}$ matrix in the following form:

$$\boxed{\lim_{t\to\infty} \boldsymbol{P}^{t} = \boldsymbol{e}\boldsymbol{\pi}} \tag{180}$$

This already implies the stationary probability distribution is independent of the initial probability distribution; and that it can be found by raising the transition probability matrix to a large power. This is however, not a practical solution as to the computational complexity of such an operation. Instead, we propose a straightforward algebraic solution. To verify this observe that the Markov transition at its limit should not change the steady state; that is:

$$\lim_{t\to\infty} \boldsymbol{\pi}\boldsymbol{P}^t = \boldsymbol{\pi}\lim_{t\to\infty} \boldsymbol{P}^t = \boldsymbol{\pi} \quad (181)$$

We also have a constraint on the values of $\boldsymbol{\pi}$, which guarantees that it is a probability distribution, or in other words $\boldsymbol{\pi}$ is an *n-simplex*:

$$\|\boldsymbol{\pi}\|_1 = \sum_{i=1}^{n} \boldsymbol{\pi}(i) = 1 \quad (182)$$

In matrix form, (recall that $\boldsymbol{e}$ is a column vector):

$$\boldsymbol{\pi}\boldsymbol{e} = 1 \quad (183)$$

Therefore, if we replace $\lim_{t\to\infty} \mathrm{P}^{\mathrm{t}}$ with the square matrix $\mathrm{e}\pi$ we can verify:

$$\boldsymbol{\pi}\boldsymbol{e}\boldsymbol{\pi} = \boldsymbol{\pi} \quad (184)$$

Seemingly, the equation (179) is 'somehow' in the form $\boldsymbol{Ax} = \lambda\boldsymbol{x}$; that is to say $\boldsymbol{\pi}$ is an eigenvector of the transition probability matrix $\boldsymbol{P}$, associated with an eigenvalue of 1. Note that we consider all vectors as column vectors, except state vectors such as $\boldsymbol{\pi}$. This is why the stationary vector is a 'left eigenvector' of the transition probability matrix, NOT a right eigenvector as in $\boldsymbol{Ax} = \lambda\boldsymbol{x}$. Had $\boldsymbol{\pi}$ been defined as a column vector then we would have $\boldsymbol{\pi} = \boldsymbol{P}\boldsymbol{\pi}$, and thence it would have been called a right eigenvector. However, then we would have to define the matrix $\boldsymbol{P}$ in such a way as to have its (i, j) entries defined as the probability of being at i with the condition of being at j before. As such a formulation would be awkward, we -as most authors do- prefer to define $\boldsymbol{\pi}$ as a row vector; and so it turns out to be a left eigenvector of the Markov matrix associated with the Perron-Frobenius eigenvalue of 1, i.e. strictly larger than other eigenvalues, whose entries are non-negative.

In order to compute $\boldsymbol{\pi}$, we rewrite the last equation as follows:

$$\boldsymbol{\pi}(\boldsymbol{I}\text{-}\boldsymbol{P}) = \boldsymbol{0} \quad (185)$$

Instead of finding $\boldsymbol{\pi}$ through simulating random walks, we have devised a way to solve this equation together with the constraint equation (183) algebraically. We will also present an algorithmic method based on the idea of Power Iteration. The algebraic solution is achieved by combining the equations through concatenating the matrices involved; i.e. specifically merging $\boldsymbol{e}$ as a column to the square matrix $\boldsymbol{I}$-$\boldsymbol{P}$ and merging a one to a zero row vector $\mathbf{0}_{1\times n}$. Formally:

$$\boldsymbol{\pi}[\boldsymbol{I}\text{-}\boldsymbol{P} \quad \boldsymbol{e}]_{n\times(n+1)} = [\mathbf{0} \quad 1]_{1\times(n+1)} \tag{186}$$

This equation looks like the following when expanded:

$$[\pi_1 \quad \dots \quad \pi_n]\begin{bmatrix} 1-p_{1,1} & \dots & \text{-}p_{1,n} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \text{-}p_{n,1} & \dots & 1-p_{n,n} & 1 \end{bmatrix} = [0 \quad \dots \quad 0 \quad 1] \tag{187}$$

Now this is almost in the form of a system of linear equations as $\boldsymbol{Ax} = \boldsymbol{b}$, which can be solved through standard iterative methods provided in Numerical Analysis or Linear Algebra libraries. To put it exactly in such a familiar form, as an implicit definition, we transpose both sides to make column vectors and a multiplication by left:

$$\boxed{[\boldsymbol{I} - \boldsymbol{P} \quad \boldsymbol{e}]^T \boldsymbol{\pi}^T = [\mathbf{0} \quad 1]^T} \tag{188}$$

However, if an explicit definition is desired then by denoting $\boldsymbol{M} = [\boldsymbol{I}\text{-}\boldsymbol{P} \quad \boldsymbol{e}]$ we can define $\boldsymbol{\pi}$ 'explicitly' using the right inverse (generalized inverse) of $\boldsymbol{M}$ denoted as $\boldsymbol{M}^R := \{\boldsymbol{U} | \boldsymbol{MU} = \boldsymbol{I}_n\}$.

$$\boxed{\boldsymbol{\pi} = [\mathbf{0} \quad 1][\boldsymbol{I} - \boldsymbol{P} \quad \boldsymbol{e}]^R} \tag{189}$$

Each of the models א, ב, ג, or ד can be plugged into these equations of course. We will now show the results of these models on our two exemplary datasets.

FIGURE 125 shows the stationary distribution of our random walk models א, ב, and ג using only the topological information (Crisp Adjacency Matrix)

FIGURE 126 shows an exemplary stationary distribution of our random walk model ד (Dalet), biased with the following distribution that is a Local Closeness Centrality. This distribution is found using a Transition Probability matrix formed using the topological information (Crisp Adjacency Matrix). Note that there can be many other types of distributions produced by model Dalet because the user can opt to use an arbitrary distribution as what they believe to be the probabilities of staying at nodes (streets).

FIGURE 127 shows a Local Closeness Centrality measured for cycling within 15 minutes on Marwell dataset, that is used as a bias for model Dalet in producing the distribution shown in the previous figure.

For undirected graphs, the models Alef, Bet and Gimel turn out to have an easy-to-find solution that is nothing but their 'normalized degree distributions'; as we had mentioned before, considering the generalized definition of degrees nodes that equal the row sums of the weighted adjacency matrix (Fuzzy Closeness Matrix in our case). We remind the reader that such matrices show how 'well-connected' are the nodes of the graph; as such, they do not contain impedance values but rather admittance values corresponding to the links of the graph understudy. The normalized degree distribution as mentioned before under the title degree centrality is nothing but the degree of each node divided by the sum of all node degrees in the graph. In case of a crisp adjacency matrix the total sum of degrees in the graph equals two times the number of links; but in case of Fuzzy adjacency matrices, this sum has to be found by simply adding up the degree values (row sums). The question that comes to mind is then why bother finding algebraic or computational solutions for the problem of finding stationary distributions of random walks. The answer is threefold: firstly, the definitions we have given above can be adapted to directed graphs as well. Therefore, the algebraic or computational; solutions will be applied in case of studying directed configuration graphs. Secondly, the model Dalet had to be derived from Alef, Bet, and Gimel for a smooth and comprehensible transition. They need to be solved algebraically or computationally using the contributed methods. The models Bet and Gimel may not strike the reader as particularly interesting models, in fact they do not produce distinct results; nevertheless, they together pave a smooth way towards model Dalet, which provides for making a wide range of stochastic models, based on real-world data, or bespoke combinations of centrality models and Random Walk models. Thirdly, as we show in the followings, the algebraic formulation (of the transition probability matrices) is used to show the state transitions using a customized Power Iteration method. This latter possibility provides for a range of interesting studies on the network, from checking ‘the probability that tourists would find a souvenir shop’ or ‘the probable whereabouts of a culprit on the run’. Now, we show here why a normalized degree distribution is a stationary distribution for a Markov Chain formed on an undirected graph by replacing:

$$\boldsymbol{\pi}_j = \frac{d_j}{\sum_{j=1}^{n} d_j} \xrightarrow{yields} \boldsymbol{\pi} = \frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{D} \tag{190}$$

Now we want to verify $\boldsymbol{\pi} = \boldsymbol{\pi P}$, hence we write:

$$\aleph: \boldsymbol{\pi P} = \left(\frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{D}\right)(\boldsymbol{D}^{-1}\boldsymbol{C}) = \frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{C} = \frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{D} \tag{191}$$

Note that $\boldsymbol{e}^T\boldsymbol{C}$ gives the column sums of the Fuzzy adjacency matrix (Closeness matrix), but if the graph is undirected then the row sums and column sums are the same and both equal node degrees; thus the solution is verified for model Alef. Similarly, for model Bet:

$$ב: \boldsymbol{\pi P} = \left(\frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{D}\right)(\psi\boldsymbol{I} + (1-\psi)\boldsymbol{D}^{-1}\boldsymbol{C}) = \frac{\psi}{2|L|}\boldsymbol{e}^T\boldsymbol{D} + \frac{1-\psi}{2|L|}\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{D}^{-1}\boldsymbol{C} \tag{192}$$

$$ב: \boldsymbol{\pi P} = \frac{\psi}{2|L|}\boldsymbol{e}^T\boldsymbol{D} + \frac{1-\psi}{2|L|}\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{D}^{-1}\boldsymbol{C} = \frac{\psi}{2|L|}\boldsymbol{e}^T\boldsymbol{D} + \frac{1-\psi}{2|L|}\boldsymbol{e}^T\boldsymbol{D} = \frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{D} \tag{193}$$

For model Gimel:

$$ג: \boldsymbol{\pi P} = \left(\frac{1}{2|L|}\boldsymbol{e}^T\boldsymbol{D}\right)(\boldsymbol{\Psi} + (\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{D}^{-1}\boldsymbol{C}) = \frac{1}{2|L|}(\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{\Psi} + \boldsymbol{e}^T\boldsymbol{D}(\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{D}^{-1}\boldsymbol{C}) \tag{194}$$

$$ג: \boldsymbol{\pi P} = \frac{1}{2|L|}(\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{\Psi} + \boldsymbol{e}^T\boldsymbol{D}\boldsymbol{D}^{-1}(\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{C}) = \frac{1}{2|L|}(\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{\Psi} + \boldsymbol{e}^T(\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{C}) = \tag{195}$$

$$ג: \boldsymbol{\pi P} = \frac{1}{2|L|}\boldsymbol{e}^T(\boldsymbol{\Psi}\boldsymbol{D} + (\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{C}) = \frac{1}{2|L|}(\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{\Psi} + \boldsymbol{e}^T(\boldsymbol{I}\text{-}\boldsymbol{\Psi})\boldsymbol{C}) \tag{196}$$

Since the graph is undirected then $\boldsymbol{C}$ is symmetric, i.e. $\boldsymbol{C} = \boldsymbol{C}^T$. We also know that $\boldsymbol{\Psi}$ is square diagonal and hence symmetric as well, therefore $\boldsymbol{\Psi C} = \boldsymbol{C\Psi}$ and so we can rewrite the last equation as follows:

$$ג: \boldsymbol{\pi P} = \frac{1}{2|L|}(\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{\Psi} + \boldsymbol{e}^T\boldsymbol{C}\text{-}\boldsymbol{e}^T\boldsymbol{\Psi}\boldsymbol{C}) = \frac{1}{2|L|}(\boldsymbol{e}^T\boldsymbol{D}\boldsymbol{\Psi} + \boldsymbol{e}^T\boldsymbol{C}\text{-}\boldsymbol{e}^T\boldsymbol{C}\boldsymbol{\Psi}) = \frac{\boldsymbol{e}^T\boldsymbol{D}}{2|L|} \tag{197}$$

We know that $\mathrm{e}^{\mathrm{T}}\mathrm{C} = \mathrm{e}^{\mathrm{T}}\mathrm{D}$ therefore the equality is verified.

The algorithmic solution for the problem of finding a stationary distribution for an ergodic Markov Chain is very similar to the problem of finding the Perron-Frobenius root (dominant eigenvalue or spectral radius) and its associated [dominant] eigenvector. The algebraic solution is elegant and can even be found efficiently, provided there are efficient numeric solvers available. A more versatile solution could be based on the concept of a Power Iteration, which is used also by Google to compute PageRank efficiently for a very similar problem, i.e. finding the probability that a random web-surfer would ever visit a page, in order to rank pages based on such probabilities. PageRank produces a ranking that helps Google bring the most important pages when somebody searches for them.

There are a few points that are not readily obvious when looking at the Power Iteration method. The first point is that in the Power Iteration method, we actually do not raise a matrix to a large power. We do the Power Iteration implicitly instead; that is we multiply the last vector iteratively by the matrix, which is effectively the same as multiplying the initial vector by the matrix when raised to a power; with the difference that the latter action is much more efficient in terms of space-time complexity. The second issue is that the normalization for finding a stationary distribution is NOT based on the Euclidean norm (2-norm); instead, it will be based on 1-norm.

One of the most efficient ways of the solution to the problem of finding a dominant eigenvector (and its associated eigenvalue a.k.a. Spectral Radius) is to run an algorithm widely known as Power Iteration. The idea of this algorithm is that if we choose a random vector and applies the transformation encoded in the matrix in question consecutively for many times, it should converge to the dominant eigenvector. The reason behind is:

- any vector in the $\mathbb{R}^n$ can be written as a linear combination of eigenvectors of the matrix in question;
- then applying the transformation matrix would be like multiplying each of these components by eigenvalues associated to the corresponding eigenvalues; and therefore
- applying the matrix transformation for many consecutive times, corresponds to finding a power series which would be dominantly determined by the term associated with the dominant eigenvector.

We have explained this through equations (105) to (113) in the section Eigenvector Centrality. Now we present our bespoke Power Iteration Algorithm for finding the Stationary Distribution of a Markov Chain.

Algorithm 10: Power Iteration for Stationary Distribution of a Markov Chain

Inputs: Transition Probability **Matrix** $[\boldsymbol{P}]_{n\times n}$ **, float** epsilon**, integer** MaxIterations

Outputs: Stationary Distribution $[\boldsymbol{\pi}]_{1\times n}$

1. Initialize RowVector Prev_Pie as RowVector[n,1];//every item is 1
2. Initialize RowVector Next_Pie as RowVector[n,1]; //every item is 1
3. Let Integer Counter=0;
4. **Do**
    - Counter++;
    - Next_Pie= Prev_Pie*P;
    - Next_Pie.Normalize(1); //p-norm 1 is used, i.e. division by sum
    - RowVector Difference = Next_Pie - Prev_Pie;
    - Let float error=Difference.Norm(2);
    - Prev_Pie=Next_Pie;
5. **While(**error>epsilon **AND** Counter<MaxIterations**)**
6. **Return Next_Pie;**

§ 5.13.4 Stationary Distributions of Directed Graphs

Previously we simplified our centrality models and calculations by assuming that the configuration graph is undirected. This was not necessary however. Here we show that the Configuration Graphs can be constructed as Directed Graphs and then show the stationary distributions of Directed Configuration Graphs found using our algebraic formulations and the iterative method described above. Firstly, we explain how the directed configuration graph is constructed. It is in fact a simple tweak in the constructor of the configuration graphs. We explained earlier in determining the angular impedance that the angular impedance, with the definition we gave is the same from i^{th} street to j^{th} street or vice versa. However, if we compute the metric impedance using the signed value of altitude difference we obtain a directed graph. Downhill have less impedance than uphill roads. It follows obviously that using this graph the pathfinding methods will work similarly but potentially give different paths from an origin to a destination and for the path other way around, provided there is a topographic landscape mesh available of course. We have such a mesh available for our Tarlabasi dataset so we display the results on this dataset.

The option to consider a Configuration Graph as directed is might or might not make a difference depending on the availability of a digital terrain model.

We avoided directed[70] graphs in previous sections for the sake of simplicity. The difference between a directed graph and an undirected graph is in the impedance values for walking or cycling. In the impedance functions used for walking and cycling, we consider that a downhill road corresponds to higher speeds of walking or cycling and up to a threshold for negative slope, below which the downhill road becomes potentially dangerous so that the walker or biker might have to reduce speed to walk or bike carefully. In reality, we might even exclude certain roads from calculations after we notice their slope degree is too steep either downhill or uphill. A directed graph naturally corresponds to an **asymmetric** transition probability matrix, therefore, we can realize that the simple generic solution (normalized degree centrality) would not be a solution for the stationary distribution; it has to be found using the algebraic or iterative solutions introduced earlier.

As seen in the example shown in Figure 128, it is clear that if the directed configuration graph shows easier access to downhill roads then the random walker is most likely to be seen in the valley as apparent in the figure. This might not strike as a very interesting result; however, it verifies that the model works mathematically and computationally. It confirms intuition. Had the model been weighted and directed differently it would have resulted in some other distribution. The threshold for Fuzzification of the impedances is 1 minute here in this model.

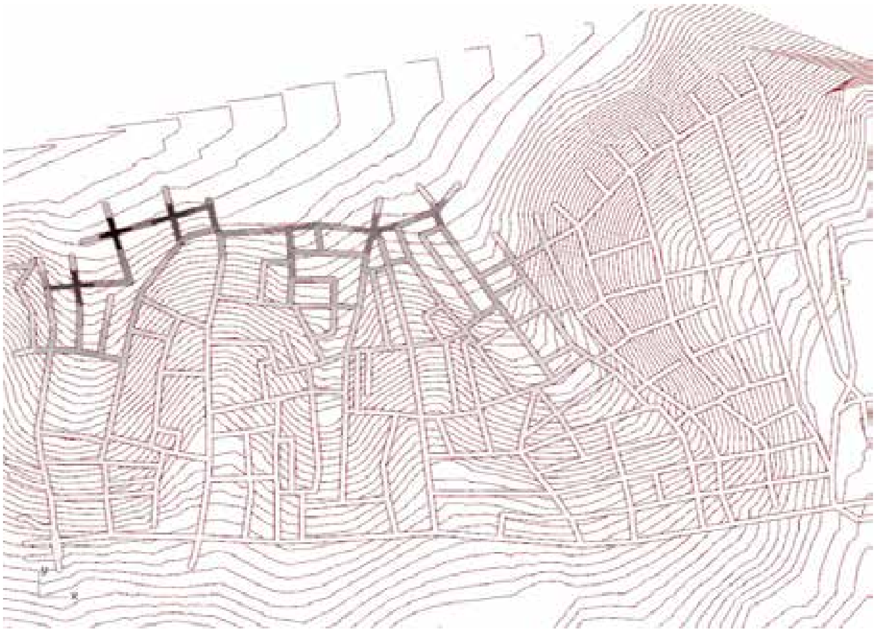


FIGURE 128 shows the stationary distribution of a directed random walk on Tarlabasi dataset model Alef, Bet, Gimel.

FIGURE 129 shows the stationary distribution of the random walk models Alef, Bet, and Gimel on a directed adjacency matrix fuzzified with a far threshold of 1.5 minutes (above which destinations are considered far and their weight of connection thus becomes zero).

FIGURE 130 shows the stationary distribution of model Dalet when applied on a topographically directed configuration graph using the staying probabilities from the distribution below (local closeness)

FIGURE 131 shows a local closeness centrality distribution used as Psi for the above model Dalet

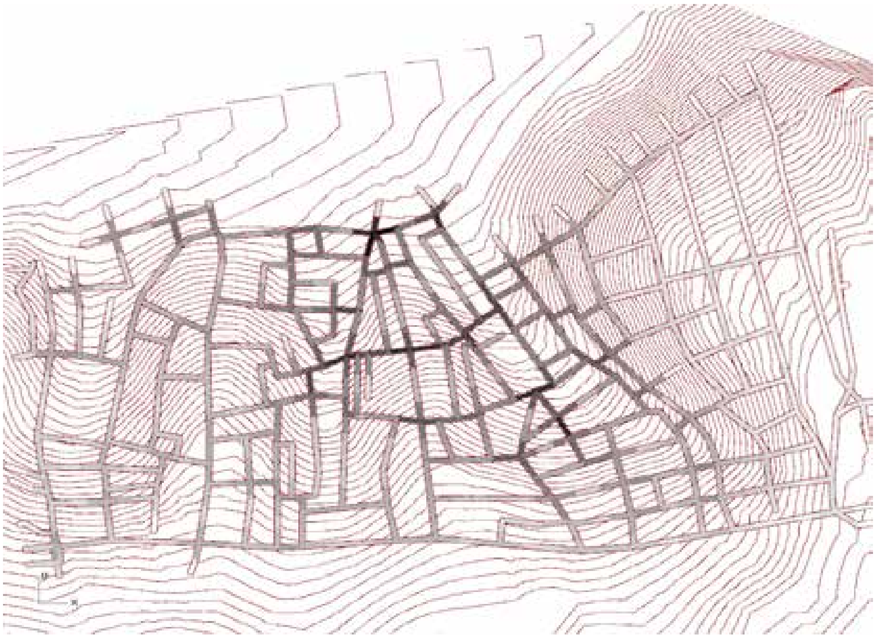


FIGURE 132 shows the stationary distribution of model Dalet when applied on a topographically directed configuration graph using the staying probabilities from the distribution below (global closeness)

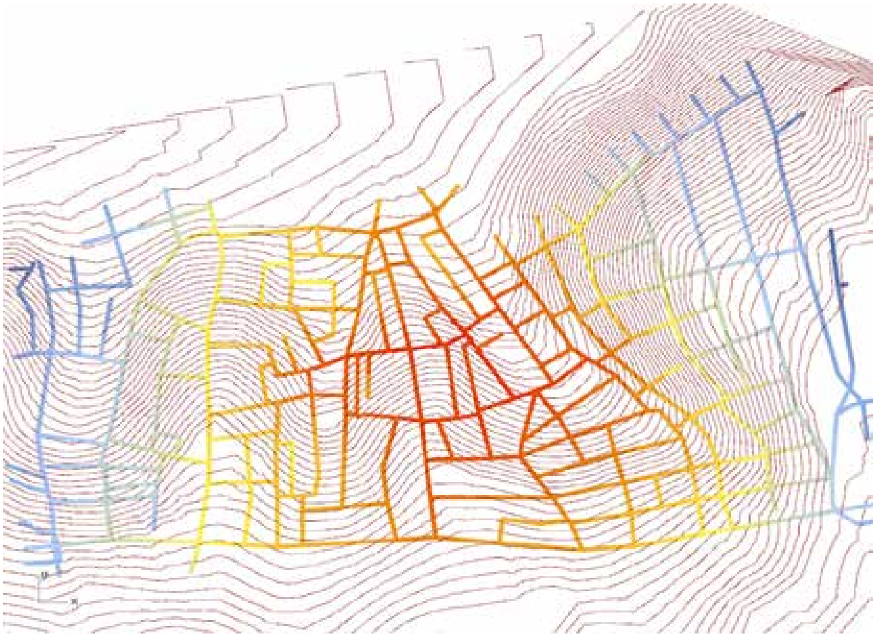


FIGURE 133 shows a global closeness centrality distribution used as Psi for the above model Dalet

§ 5.13.5 Culprit On the Run and the Souvenir Shop

In order to illustrate the functioning of a random walk model we metaphorically show an example of a culprit last seen at a point, where the police wants to know the probable whereabouts of him after say 10 minutes, to locate their resources efficiently. Suppose that either there is no transit exit point where the culprit in question can go to or those points are already being controlled; i.e. there is no possibility for the random walker to 'teleport' to some other place in the city. If the random walker starts a random walk in a certain street, then their probable whereabouts can be determined as the transient states of the Markov Chain representing the transition probabilities of the random walk. We first show an algorithm that computes the transient state vectors; then show a hypothetical example of the probable whereabouts of a culprit on the run using a random walk of type Alef; after that, we show an exemplary application of a random walk model of type Dalet in finding a good location for a hypothetical souvenir shop.

Algorithm 11: Power Iteration for Finding Transient Distributions of a Markov Chain

Inputs: Transition Probability **Matrix** $[\boldsymbol{P}]_{n\times n}$**, float** epsilon**, integer** MaxIterations**, integer** StartingNode
Outputs: Transient Distribution $[\boldsymbol{\pi}]_{1\times n}$

1. Initialize RowVector Prev_Pie as RowVector[n,0];//every item is 0
2. Prev_Pie[StartingNode]=1;//random walker is seen certainty first at this location
3. Initialize RowVector Next_Pie as RowVector[n,1]; //every item is 1
4. Let Integer Counter=0;
5. Do
• Counter++;
• Next_Pie= Prev_Pie*P;
• Next_Pie.Normalize(1); //p-norm 1 is used, i.e. division by sum
• RowVector Difference = Next_Pie - Prev_Pie;
• Let float error=Difference.Norm(2);
• Prev_Pie=Next_Pie;
6. While(error>epsilon **AND** Counter<MaximumIterations**)**
7. **Return Next_Pie;**

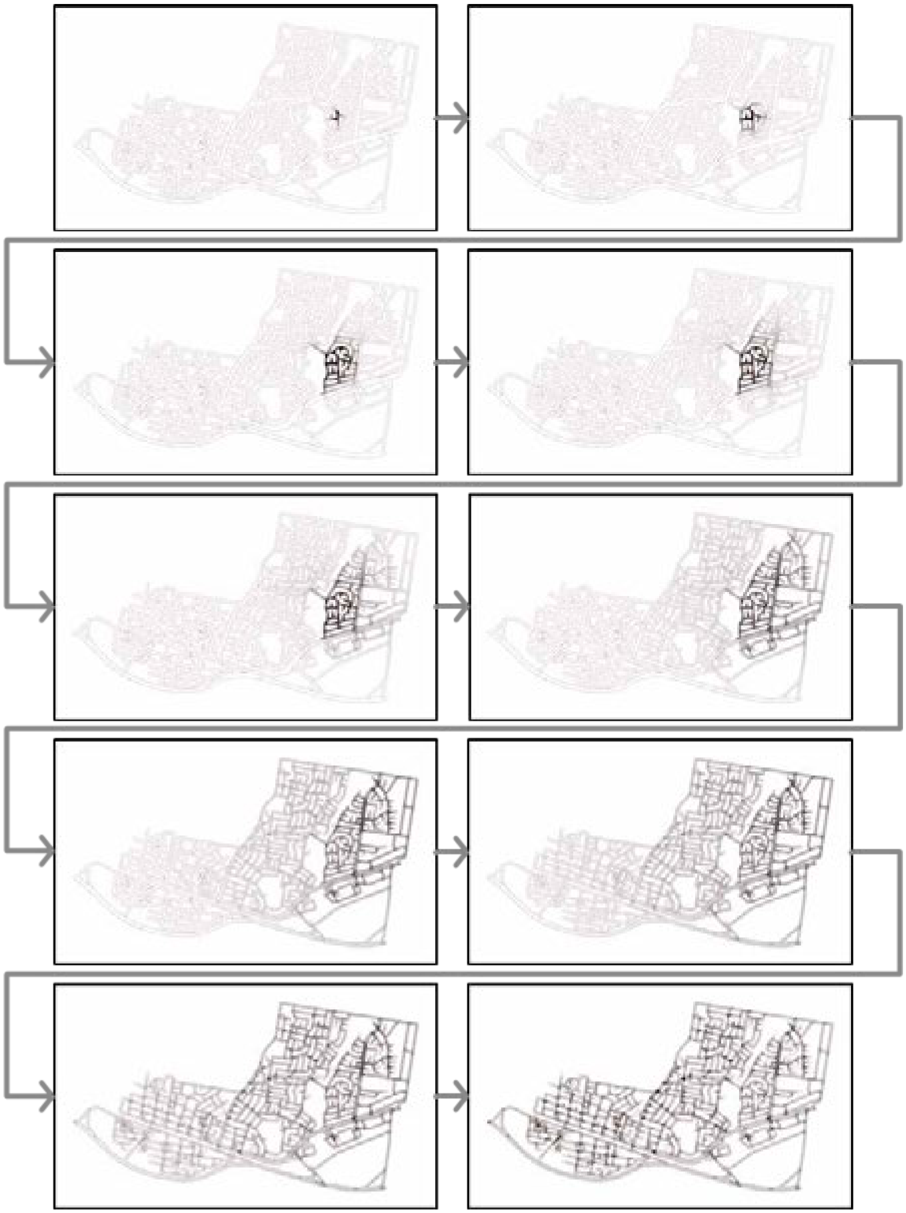


FIGURE 134 shows the probable whereabouts of an imaginary culprit on the run; that is the transient states of a random walk progressing towards an ultimately stationary distribution.

Here we provide another example could illustrate the potentials of our Random Walk models. Suppose we want to establish a souvenir shop in a touristic town. Our best chances of prosperity would be in places where tourists are likely to pass by. Assuming tourists as random walkers that are likely to visit a number of POI and then wander around, then we can use our random walk model Dalet to map the probability distribution of passage of such toursist. In this exemplary application we do not consider the influence of such things as street signage and touristic maps or smartphone applications.

FIGURE 135 shows a number of hypothetical POI in Marwell and the Fuzzy closeness centrality towards all of them. We have used this distribution to bias the probabilities of staying at streets for a random walk model of type Dalet. See the next figure.

FIGURE 136 shows the stationary distribution of a random walk started at the node marked with a red dot by an imaginary tourist. We shall see in the next figures that regardless of the starting point, model Dalet puts out the same stationary distribution for such walks.

FIGURE 137 shows the stationary distribution of another random walk started from the street marked by a red dot, which has eventually converged to the same stationary distribution. This distribution is biased as to the Fuzzy closeness to POI 0, 1, and 2.

FIGURE 138 shows the stationary distribution when the random walker might have started their journey from any node with a probability of 1/n.

§ 5.14 Future Work (Limitations and Open Problems)

The main limitation of our work on modelling pedestrian accessibility is that we consider the pedestrian movement to be bound to the network space, whereas in reality pedestrians are freer to choose alternative ways to go through. Using a raster representation (2D pixels or 3D voxels) of the navigable public space we could theoretically compute the same measure for the network space represented with raster nodes as well; however, this would not be feasible for most practical applications in urban design and planning due to the huge computational load of graph algorithms on such models. Note that we are computing all optimal paths, which is of time complexity $O(n^3)$, that is by doubling the size of network the worst case computation time would be multiplied by a factor of $2^3 = 8$. Using datasets such as OpenStreetMap data for pedestrian networks worsens this issue because in many cases, some important pedestrian paths might be missing and in public squares, the paths might not be good representatives of actual movement trajectories. A workable strategy to overcome this issue at a theoretical level would be implementing a Topological Straight Skeleton tool as introduced in the section on Spatial Network Models.

Our solution to the problem of validating the topology of a street centreline network is not a standardized scalable solution yet. This issue of cleaning and validating the topological structure of such datasets is not in the scope of this research but in reality affects and impedes any practical experimentation with network studies. Fully automated solutions for correcting dataset problems would not be workable or even desired because human interaction would be required in some cases to ensure the representativeness of the network data models.

Our Easiest Path algorithm as presented in this book is not yet extendable to encompass other aspects of optimality such as pleasance or safety. This is a theoretical limitation, which we have accepted to ensure the physical correctness of our model, in terms of the commensurability of the travel costs. An idea for future work is to develop an alternative extendable model that could encompass many aspects in path finding without compromising the physical meaning of the models.

Our defuzzification method only works for a single farness threshold. If we were to interpret (defuzzify) a fuzzy aggregate of multiple values each of which fuzzified with different thresholds, then we could only do it with a single threshold. The question on how to aggregate the initial fuzzification thresholds for defuzzification is still open.

Remarkably, our geodesic centrality models would not be disturbed with network edge effects because we can always consider a buffer area exactly larger than or equal to the fuzzy threshold used for marking far destinations. However, the spectral models introduced need to be studied in terms of their ranking stability under network edge effects.

The Markov Chain models theoretically consider the mobility behaviour as being stochastic, whereas the geodesic centrality models consider the same behaviour to be deterministic. The actual mobility behaviour seems to be in the middle of these poles and so we need to develop a spectrum of alternative models for explaining the actual mobility behaviour.

We are aware that the actual decision-making of people in way finding is more complicated than being based on our easiest path and is affected by perceived safety, security, pleasance, road quality and perhaps attractions such as shops throughout routes they might take. Besides, we are not claiming to have made a tool for predicting actual walking or cycling patterns with our Easiest Path algorithm. We believe that considering those additional parameters requires deeper methodological research accompanied by empirical research into actual movement patterns obtained through web/mobile applications that collect GPS tracks of people. For this purpose, we believe such data of tracks needs to be aggregated temporally and spatially in order to be representative of the actual movement patterns. In absence of such data, previously, researchers have used data sets that only include hours of movement data collected. Given the availability of reliable unbiased data over long terms and many cities, we see a great potential in researching actual movement patterns. Many people are sceptical about predictive urban models, perhaps rightly so, however, we believe that such models are needed to provide insight into planning actions. Every model will be wrong in the absolute sense of the word but we can think of models that can better explain the complex behaviour of people in built environment and thus come useful in assessing plans.

As is the case in most spatial analysis researches, we can only analyse what is meaningfully and soundly representable on a map. That is to say, we do not deal with such things as beauty or safety of a route, for we do not have a rigorous way of measuring them. Therefore, the methods provided can be deemed as describing the potentiality of movement but not its actuality.

We have not directly addressed land-use effects on mobility. However, in reality land-use has a very important role in shaping mobility patterns. In situations like vernacular or organically grown towns land-use distributions might match very closely to the network centralities but there are many mismatches in new towns and there is such a thing as land-use attraction effect that could affect pedestrians or cyclist flows. This is the main area of work that we find worthy of methodological attention for our future work.

§ 5.15 Conclusion (Summary of Achievements)

The models and methods presented in this chapter together form a conceptual framework for spatial network analysis and lay a foundation for constructing a class of probabilistic models that can address the complexity of way finding in urban areas. We hereby enumerate the key achievements and major theoretical or technical limitations of our work.

The Easiest Path algorithm finds a path for walking or cycling that is as 'simple, short, and flat' as possible. Simplicity of an Easiest Path can be controlled with a parameter that indicates confusion because of turning away; the time spent to figure out which way to go next. Using this algorithm, we define temporal distance between locations as the minimum time it would take one travel from an origin to a destination.

The Easiest Path algorithm is both mathematically and physically valid. The mathematical validity stems from the fact that the dynamic programming methods used (such as Dijkstra or Floyd-Warshall algorithm) are guaranteed to find the minima. The physical validity is consolidated because we ensure the commensurability of variables to be added together. We measure cognitive and physical traversal impedance in terms of travel time and therefore they both have the same physical dimensions.

The Easiest Path algorithm relies on our novel 3D Configuration Graph model of spatial networks that allows for attributing length as well as elevation and direction change to links. It is a Street-to-Street or Line2Line model that allows us to achieve an integration of physical and cognitive impedance in a physically sound way while lowering the computational complexity of finding simple paths or angular shortest paths by a factor of 8. This is because of the fact that the only alternatives available to our Easiest Path algorithm [Simplest Path of (Duckham, M., and Kulik, L., 2003) or Angular Shortest Path of (Turner, A., and N. Dalton, 2005)] require constructing a network out of doubly directed edges doubling the size of network and thus raising the time complexity of finding all optimal paths by a factor of 8 because the best algorithms for finding all optimal paths have a complexity of $O(n^3)$.

Our 3D Configuration Graph model of spatial networks also easily allows building directed graphs representing the relative ease of going along downhill roads for pedestrians and cyclists. In studying multimodal mobility patterns or such plans as Bike Sharing infrastructures having the ability to see the network and geodesics as directed would be a great advantage.

The Fuzzy accessibility analysis framework introduced puts an intuitive notion of closeness in mathematical form and facilitates a wide range of queries that can be run with intuitive notions. The fuzzy accessibility models introduced (proximity and vicinity) can be understood as fuzzy catchment models. Using this Fuzzy methods a planner or even an lay citizen can run enquires into matters such as good walking or cycling access to a combination of points of interest. One can easily find out if the new apartment they want to buy has a good access to a supermarket a transit station nearby and a school. The methods can thus be used in potential real-estate applications like WalkScore[a] for helping people find living places with high active mobility potentials. Such awareness on massive scale can potentially cause a force towards pedestrian/cyclist friendly developments in cities. The zoning methods provided give accurate insights on the accessibility of amenities for pedestrians and cyclists and therefore can provide valuable business intelligence for the retail sector. We have provided a consistent framework for measuring walking and cycling accessibility in terms of temporal distance to all, any or some points of interests. We can claim that our fuzzy closeness nearly represent human perception of distance -while being mathematically and physically correct in that they model nearness in terms of temporal distance given easy access to locations. They simply reveal what they say, take for instance the examples investigating whether residences have a reasonable 10 minutes walking or cycling distance to any grocery store; or whether they have walking or cycling access to a grocery store, a train station and a school; and if so, how good is their access? These are the types of questions the tools reliably answer taking into account the physical and cognitive realities of walking and cycling.

Our centrality models are more inclusive than their counterparts (such as those of Space Syntax) are; and they are easier to interpret and understand. They have been explained and justified with detailed mathematical and computational formulation. Our local closeness model in particular resembles the local integration in Space Syntax but encompasses two types Space Syntax models (topological integration and angular integration) and adds another component that is the topography. It does this inclusion while remarkably making the whole model much easier to interpret, understand, and explain. Our centrality models exactly model what they say they model. They do not require lengthy philosophical explanations and justifications. This has been our intention to avoid such complications as much as possible. For this very reason our models are falsifiable that is they can be shown as not corresponding with reality. However, they truly reveal what they are supposed to represent that is the potentials of the built environments in terms of walking and cycling mobility (betweenness) and accessibility (closeness).

The Random Walk models introduced are unique in that they provide a family of models with an extensive and in depth explanation that is unprecedented in the area of built environment analysis. The Random Walk model Dalet can be reliably biased

a https://www.walkscore.com/

with external inputs that could present the chances of stating at nodes. This model can also be seen as having applications outside the area of this research. We have provided verified algebraic and algorithmic solutions for finding the stationary distributions of these models.

Centrality measures and probability distributions will be interesting when deemed as indicators of some kind of activities. However interesting that prospect might seem, further data intensive research would be required to validate such capabilities. The algorithms are presently only implemented for a parametric CAD environment but we plan to provide them as cross platform web-based applications so that they could be used in routing applications as well as business intelligence applications or as spatial decision-support systems for providing insight on walking or cycling mobility and accessibility in design and planning.

6 Implementation & Test B: CONFIGURBANIST

In this chapter, we present a systematic line of work in implementing, testing, and improving urban configuration analysis models and method for studying walking and cycling accessibility. We have implemented the models, data structures, and algorithms introduced in the previous chapter as a computational toolkit called CONFIGURBANIST, written in C#.NET. The toolkit has undergone three major revisions since 2012 after being tested in three workshops in Istanbul (Tarlabasi Datascope, June 2013), Delft (31st eCAADe, September 2013), and Vienna (33rd eCAADe September 2015). The toolkit has been released publicly since September 2013. The kernel of the toolkit is a library of computational methods that is to be merged with those of SYNTACTIC and released as a Dot NET[a] library called **configraphics.dll**.

This chapter introduces the implementation of CONFIGURBANIST urban configuration analysis methodology as follows:

- Expresses the goals and outlook of the toolkit as to its target users;
- Gives an overview of the guiding principles for its UI;
- Shows the structure of the methodology;
- Introduces the tools in the implemented toolkit;
- Discusses some specific implementation details and issues;
- Explains the architecture of the configraphics.dll;
- Clarifies our stance on verification and validation;
- Reports the test results and their statistics on a sample dataset;
- Shows a process of generating accessibility evaluation reports;
- Concludes by providing a qualitative evaluation of the toolkit; and
- Discusses future development strategies for CONFIGURBANIST.

a Microsoft Don NET framwork: http://www.microsoft.com/net

§ 6.1 Introducing CONFIGURBANIST: a toolkit for urban configuration analysis

The toolkit CONFIGURBANIST[a] (presented in Figure 50) is a plugin application consisting of tools for the graphical-algorithm-editor[b] Grasshopper3D©[71]. Early versions of this toolkit were inspired by SpiderWeb[c], developed by Richard Schaffranek, i.e. a toolkit for Grasshopper providing graph theory algorithms for configurational analysis and synthesis. The current version is written in C#. It uses matrix datatypes and algorithms from MathNet[d] library for scientific computing[72].

FIGURE 139 shows the logo and the appearance of the CONFIGURBANIST toolkit, last publicly released version, as in Grasshopper3D© environment, version 9.0076, 2015; Cheetah the CONFIGURBANIST by Pirouz Nourian & Samaneh Rezvani is licensed under a *Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License*. Based on a work at *https://sites.google.com/site/pirouznourian/configurbanist*. Permissions beyond the scope of this license may be available at *http://www.grasshopper3d.com/group/cheetah*.

There have been several cycles of test and development for producing CONFIGURBANIST, out of which two stable versions have been released publicly. The last version was tested in Cityscape Configuration workshop (33rd eCAADe conference in TU Wien, Sep. 2015). In this chapter, we give a brief overview of the underlying structure of methods implemented in the CONFIGURBANIST toolkit.

---

a The toolkit CONFIGURBANIST has a dedicated user-group currently having 102 users worldwide: http://www.grasshopper3d.com/group/cheetah

b NOTE: the modules shown in this chapter are made for Grasshopper3D, but they are NOT made up of Grasshopper3D components, or any of its plugins. We have not used any graph algorithm libraries either. The code behind the modules (the CONFIGURBANIST plugin) is currently about a few thousands of lines long, written in C# (formerly in VB.NET) within Visual Studio. The environment of Grasshopper3D does not show the direction of data transmission in wires but it is always from left to right. Knowing the direction of wires, then a GH file is exactly a structured flowchart, which performs computation. However, the computation modules are all made up of Microsoft .NET Framework languages (VB.NET or C#) and in some cases made up of Python code snippets.

c http://www.grasshopper3d.com/group/spiderweb

d http://www.mathdotnet.com/ an open source DotNET library for scientific computing

§ 6.2 Goals, Outlook and Target Users

The main goal behind developing CONFIGURBANIST has been to realize, test, and improve the methods introduced in the previous chapter. It is projected that the toolkit will be transformed to a web-based application that will be useable for all who are interested in assessing walking and cycling accessibility potentials. The ultimate target users are those whose decisions might have an effect on walking and cycling accessibility in cities and new housing projects. To this end, it is argued that a useful tool is one that is useable by many, not only professionals. While it is ideal to have such a tool in such a widely accessible environment (e.g. a web-based tool); the very formulation of a set of desirable functionalities and features for such a tool is a challenging design task, which is naturally iterative.

Matters of efficiency and efficacy, as well as methodology and technology are often mistaken with one another. There was no structured definition of the tool available prior to experimentation. The methods and techniques have been therefore developed jointly along the course of the project. Before jumping into matters such as improving the efficiency of algorithms, the efficacy of the methodology as a whole has been the focal point. The technology used for developing the toolkit (i.e. Dot Net components within Grasshopper3D environment) might not be ideal regarding the abovementioned target users. However, it provided for an agile development strategy and several rounds of rapid development, test, validation, reformulation, and improvement. The fact that the Grasshopper3D is a 'Flow-Based-Programming' environment makes it an ideal platform for combining different tools and approaches. For this reason, any alternative technology is desired to allow for such a convenient and understandable approach as well.

Urban design and planning are ever more becoming about thoughtful intervention rather than creating something from scratch. For this reason, it is of outmost importance to be able to measure or estimate the effect of a change in a situation. Such changes are usually simulated in settings usually referred to as 'what-if scenarios'. An integral part of measuring the effect of design/planning scenarios is analysing built environment and measuring its performance.

Suppose we want to 'measure' the quality of a neighbourhood in terms of how good is access of people on foot/bike to a number of destination that are important on a daily basis. In this case, we have to be able to answer questions such as below:

- How favourable is walking or cycling access to an important location for residences in a neighbourhood?
- How good is the access of a location to all/any of important destinations?

Before answering the above questions, we should answer a more fundamental question that is "what is the easiest way to get from an origin to a destination?" This is to find a basis for measuring distances as 'actual distance' or 'experienced distance', i.e. the (spatial or temporal) length of a geodesic (an optimal path). Geodesics are longer than straight lines in urban environments and therefore the notion of distance should be re-defined based on geodesics. There are the two approaches for studying the relation of walking and cycling mobility to the structure of built environment through geodesics, namely, Transportation Planning and Spatial Analysis. Each of these two groups of models have their strengths but they do not model the entirety of path finding for walking and cycling in that they either disregard the cognitive aspects such as ease of navigation (typical in transportation models) or the physical aspect such as distance and steepness of routes (typical in spatial analysis models). We provide an alternative method of combining physical and cognitive impedance into path finding problem in a 'physically sound' way, which directly leads to a consistent definition of 'experienced temporal distance'. Using this method, we define a number of accessibility measures that are directly understandable for urban planners as well as non-professional citizens.

§ 6.3 Designer-Computer Interface

The process put forward by the toolkit proceeds as follows:

1. Simplify, generalize and validate the topology of a street centreline network;
2. Construct a topological model (a dual graph) from a street centreline network;
3. Search the graph for 'easiest paths' that minimize physical and cognitive travelling effort, both of which measured in terms of time and compute the 'temporal distance' of locations from each another;
4. Compute Polycentric Accessibility measures towards a number of POI;
5. Translate distances to Fuzzy Closeness;
6. Answer the questions such as "how close is an origin 'to all destinations' or 'to any destination' of interest?"
7. Divide the neighbourhood into zones of preferred access to a number of destinations (e.g. grocery /convenience stores);
8. Find most important routes in each zone;
9. Compute Geodesic Centrality indicators, i.e. Closeness Centrality and Betweenness Centrality using Easiest Paths;
10. Compute Spectral Centrality measures (Eigenvector Centrality and Katz Centrality);
11. Compute Random-Walk measures (Markov Chain models Alef, Bet, Gimel, and Dalet);

We have had a number of guiding principles in developing our design methodology regarding communication with the user (designer), in light of our general usability goals.

- The method should demand as little information and parameter values as possible and be as generic as possible;
- While simplifying the process, also allow for interactivity and the possibility of reconfiguring workflows wherever possible;
- Do not provide automatic assessment; instead provide the tools to for making an assessment of accessibility, i.e. only measurements but no automated judgement.

§ 6.4 The Urban Configuration Analysis Workflow

Our first implementation of the toolbox CONFIGURBANIST (Nourian, P, Sariyildiz, S, 2012) was radically different (much simpler) than what is represented in the previous chapter, namely the accessibility analysis methods "proximity" and "vicinity" were based on arbitrary weights and there was no Fuzzy Logics framework, also there was no Easiest Path algorithm.

§ 6.4.1 Directed Graph from Doubly Directed Streets

Our next implementation of the toolkit after inventing the Easiest Path algorithm was based on a 'doubly directed street network', as explained in the previous chapter. This implementation was in a way very precise in the sense that its Easiest Paths are guaranteed not to have detours. However, this rather perfectionist approach comes at a high cost of computation as well as a great deal of complication in development and test. The subtle issue is that if we compute Easiest Paths on doubly directed edges then for each street we will effectively have attributes for graph nodes (which will be oppositely directed edges of streets); but we are interested in attributing only one value to each street, be it distance, closeness or any centrality index. This requires to identify the two nodes (doubly directed edges in the street network) corresponding to each street in a lookup table. For future reference, we give an overview of what it would take to compute everything based on a doubly directed street network, i.e. constructing and maintaining a construct that we call an Easiest Path Map (some kind of a lookup table). While finding paths, e.g. in producing a path as a sequence of nodes or in the course of computing Betweenness Centrality, we had to work with *four paths per each OD pair*. This is because a geographic origin or destination would correspond to a street, which is itself representable by no less than two nodes (directed edges) in the network. The combinations of these alternative OD pairs correspond to 4 Easiest Paths, out of which actually one would represent the path that is effectively optimal, i.e. the path out of four alternatives with the minimum costs. This is similar to the approach used by the pioneers of finding cognitively easy/simple paths, namely Angular Shortest Path (Turner, A., and N. Dalton, 2005) & Simplest Path (Duckham, M., and Kulik, L., 2003).

We did not publicly release this version of the toolkit (reported in (Nourian, P, van der Hoeven, F, Rezvani, S, Sariyildiz, S, 2015)) as it seemed too complicated for its potential users. Here we give an overview of the structure of the methodology based on doubly directed street networks. We have decided to discontinue this approach, however we assume it is still theoretically valid, though not effectively practical in dealing with real-world networks. Besides, implementation and maintenance of the code will be much more difficult compared to our current approach.

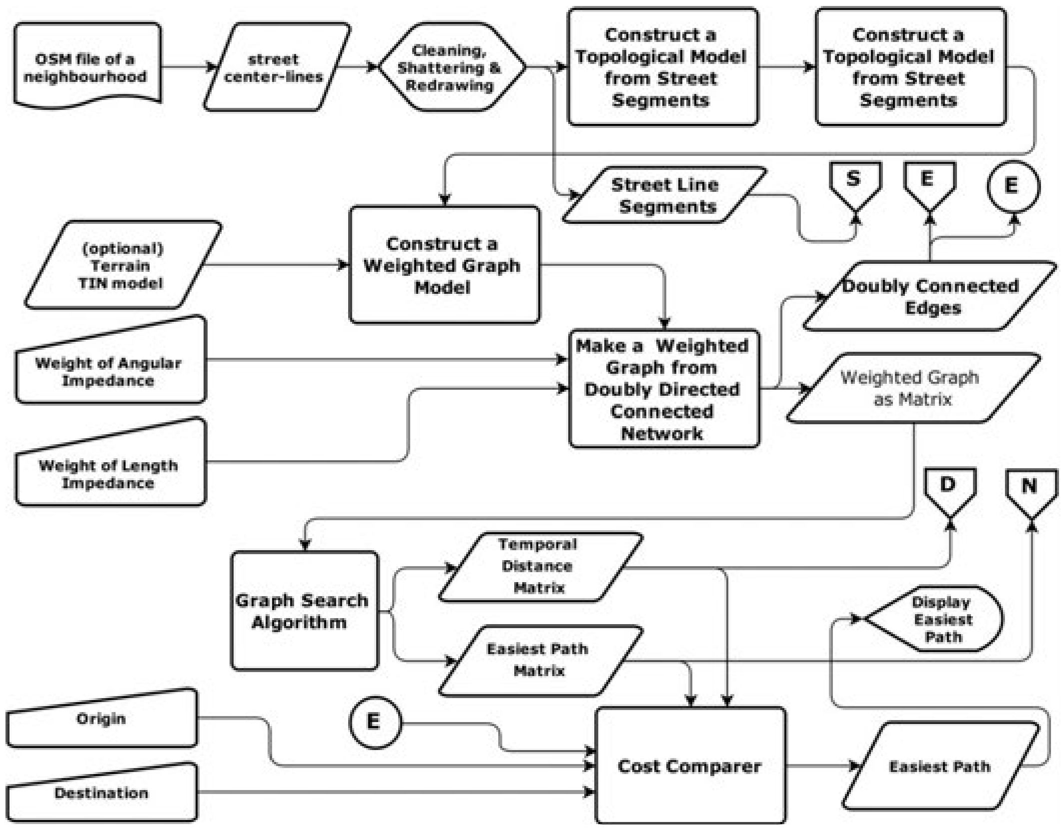


FIGURE 140 (Part 1) Our previous workflow using a directed graph on a doubly directed network (Nourian, P, van der Hoeven, F, Rezvani, S, Sariyildiz, S, 2015): our current proposed workflow is simpler because it is based on a dual directed graph model on an undirected network, not a doubly directed network.

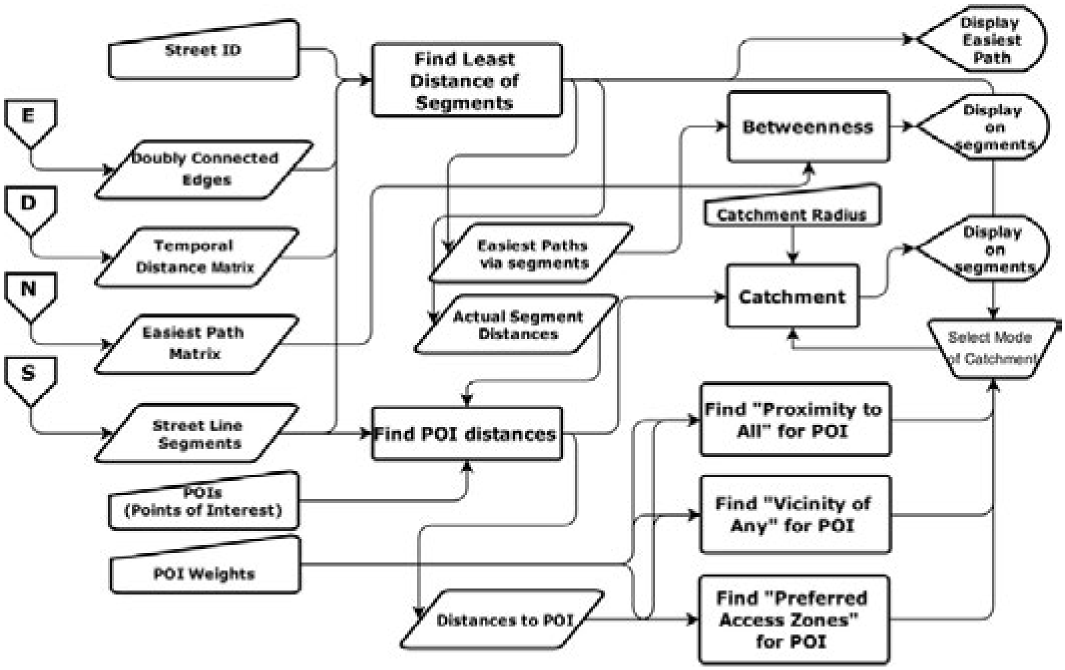


FIGURE 141 (Part 2) (Nourian, P, van der Hoeven, F, Rezvani, S, Sariyildiz, S, 2015)

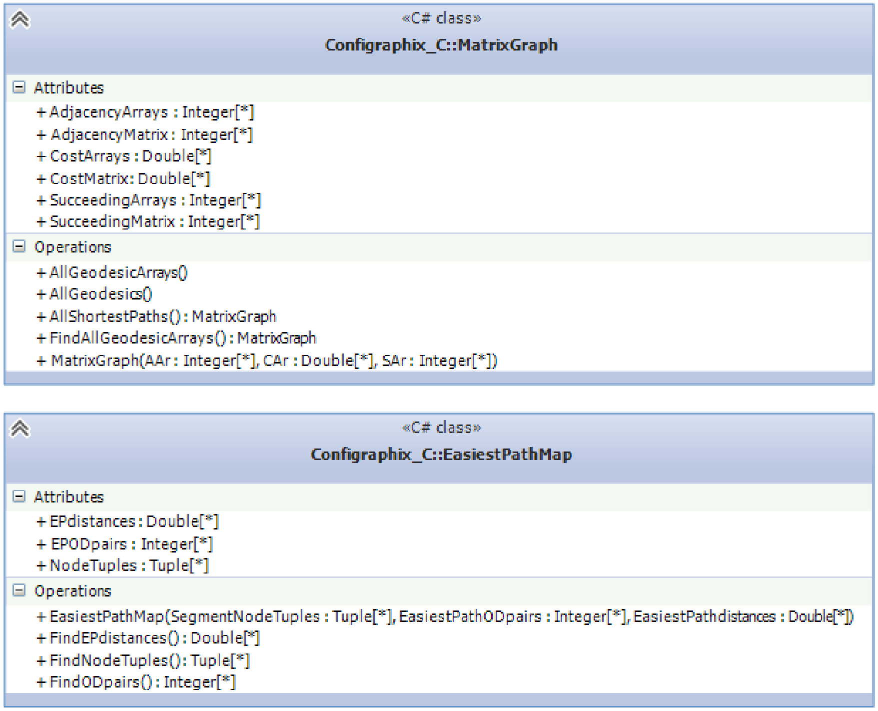


FIGURE 142 the structure of a DLL made for computing Easiest Paths on Doubly Directed Networks

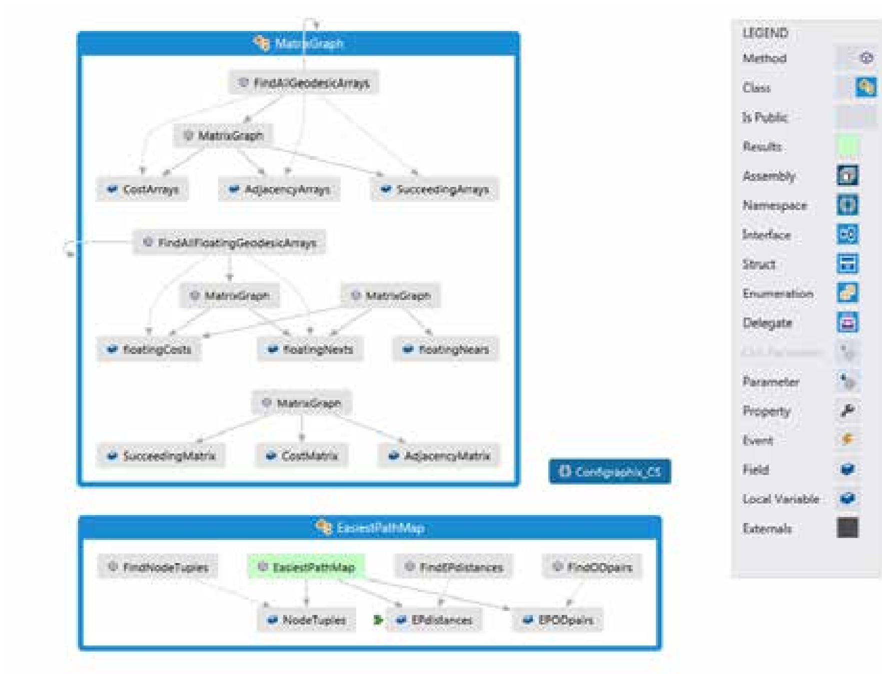


FIGURE 143 the architecture of Configraphics_CS.dll, containing methods for computing Easiest Paths on doubly directed street networks. This library was tested and verified; but we have discontinued developing it in favour of our new implementation that is easier to use. We provide descriptions of this implementation merely for future reference. The new library is introduced in section 6.7.

§ 6.4.2 Directed Graph from Undirected Streets

The latest version of CONFIGURBANIST implements the Easiest Path algorithm on a Directed Graph constructed on a network of undirected streets. This simplifies the workflow compared to the previous implementation using doubly directed street networks. We explicate the workflow in two flowcharts below. The flowcharts do not show some details and utility tools such as Graph Drawing tools and alike. For a more detailed view of the methods, see the next section on the tools themselves. The first flowchart shows the steps required prior to Graph Traversal (finding Easiest Paths and Temporal Distances); and the second flowchart shows how the results of the first part are used in constructing accessibility and centrality measures.

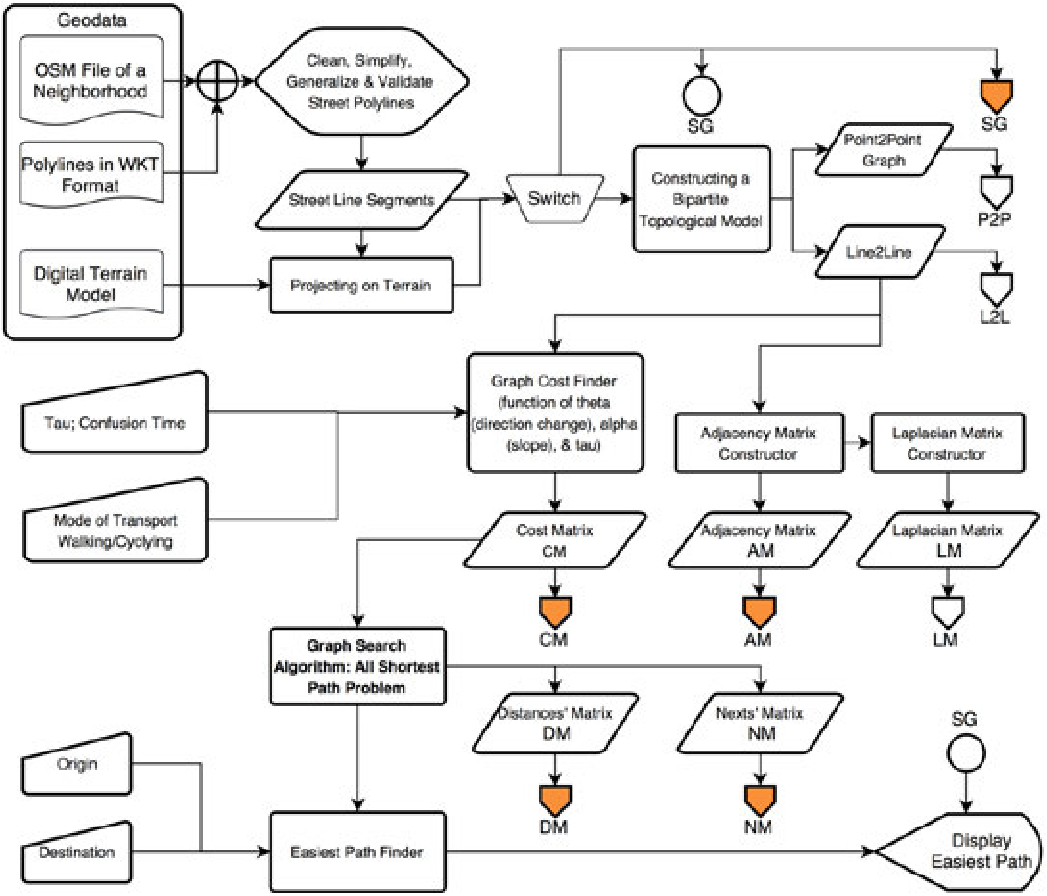


FIGURE 144 flowchart model of CONFIGURBANIST methodology (part one)

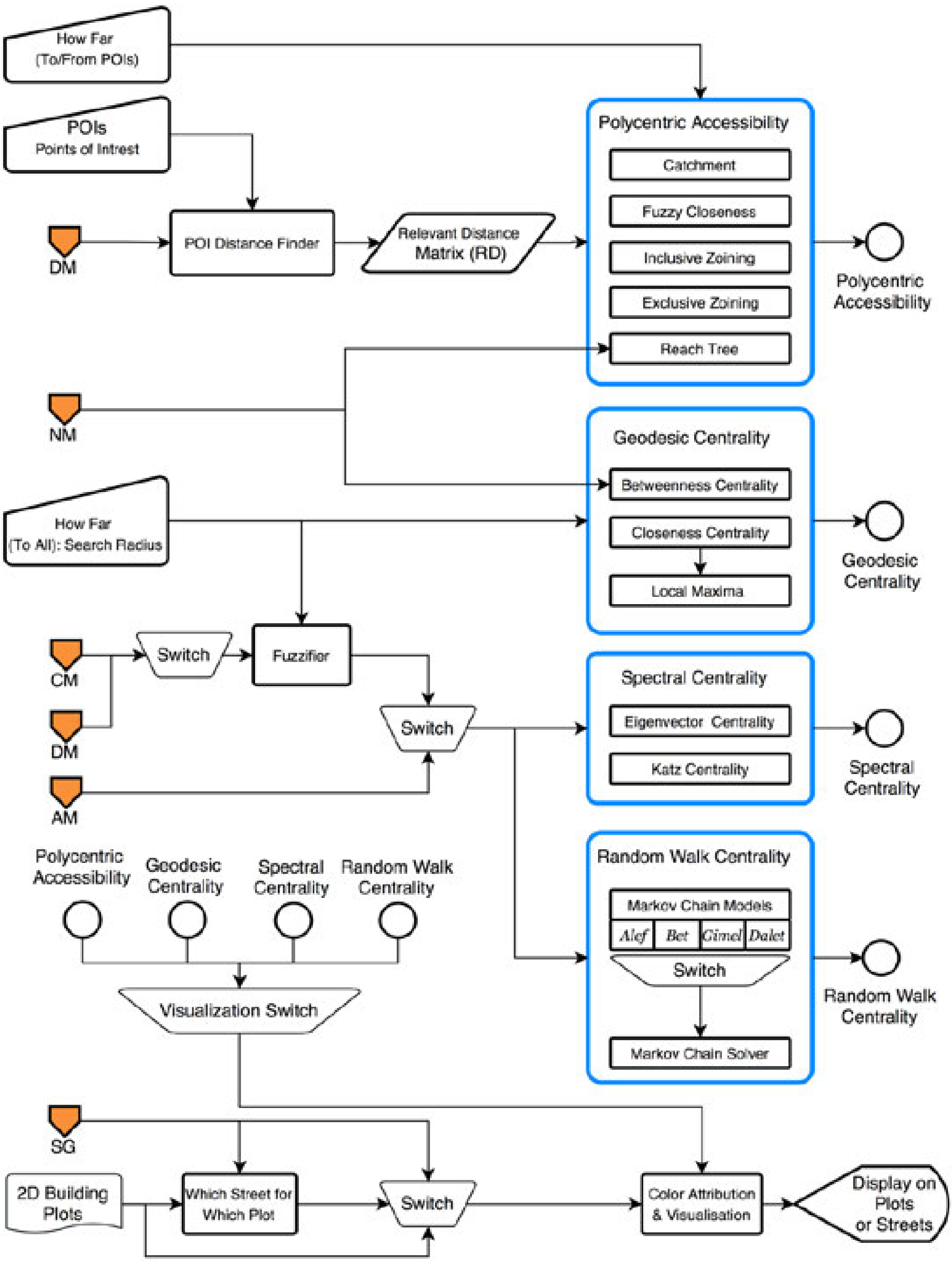


FIGURE 145 flowchart model of CONFIGURBANIST methodology (part two)

§ 6.5 Tools

The toolkit provides several groups of tools as shown in Figure 146, Figure 147, Figure 148, Figure 149, Figure 150, Figure 151, Figure 152, Figure 153, Figure 154, Figure 157, Figure 155, and Figure 156. Note that all wires are directed from left to right.

§ 6.5.1 Map Simplification and Generalization Tools (Figure 146)

Prior to constructing an abstract connected graph representation of a street network, we need to ensure the validity of the geographical representation. This very first step can be very time consuming and challenging, especially when relying on Volunteered Geographic Information (VGI) sources such as OpenStreetMap (OSM). The steps required ensuring the validity of a representation, simplification, and generalization[73] cannot be easily automated, as they usually require some user input. We offer two sets of tools for this purpose; the set of simple tools is represented in Figure 146. We shall present a more sophisticated process in explaining the steps required for constructing Network topological Models.

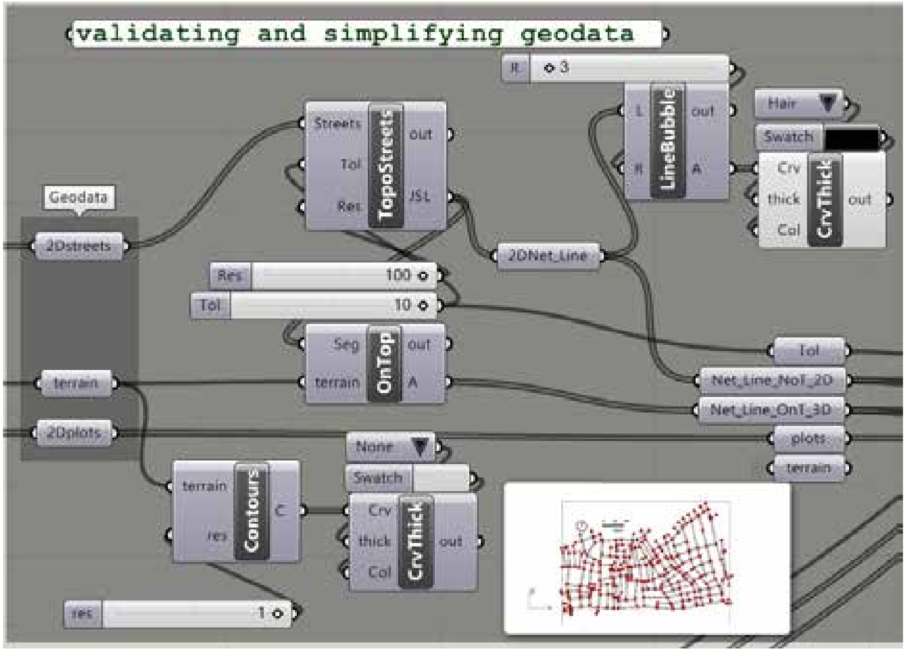


FIGURE 146 a simple set of tools for topological cleaning of a street network, shattering it at a desired resolution, simplifying it, and projecting it on a topographic terrain (if provided).

§ 6.5.2 Topological Modelling Tools (Figure 147)

This set of tools makes the process of graph construction more manageable and transparent. The Bipartite Adjacency matrices are made using the Point-to-Line Incidence Matrices ($\boldsymbol{A}^{VE}$ & $\boldsymbol{A}^{EV}$) as introduced in the previous chapter. It could be that for some reason, such as higher precision (not accuracy) in measuring distances, a Point-to-Point Graph is preferred to a Line-to-Line Graph. In that case the user can choose the P2P ($\boldsymbol{A}^{VV}$) output from the bipartite adjacency matrices P2P and L2L ($\boldsymbol{A}^{EE}$). Besides, the P2P adjacency matrix can be used in systematic generalization of a street network, e.g. in finding Connected Components as shown in “Network Topological Models”.

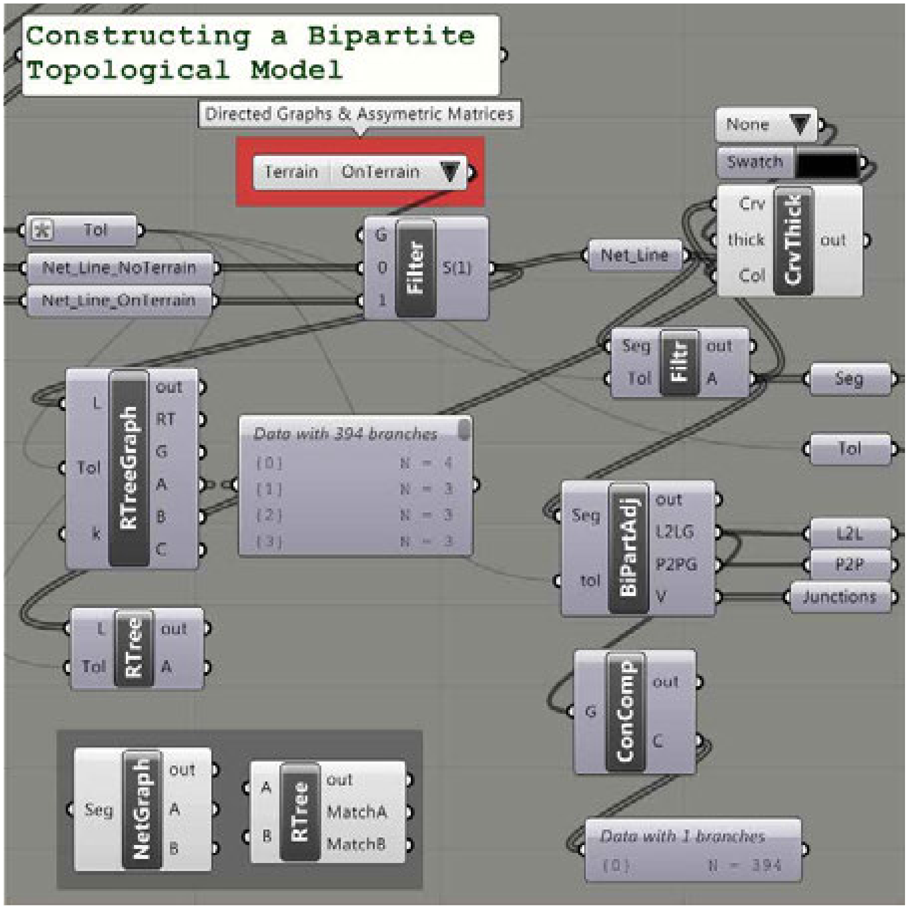


FIGURE 147 Topological modelling tools, including components for constructing an RTree spatial index, constructing Bipartite Adjacency Matrices (A^{EE} and A^{VV}) out of Incidence Matrices ($A^{VE}$and A^{EV})

§ 6.5.3 Graph Construction and Graph Traversal Tools (Figure 148)

The first key component in this set of tools initiates the Line-2-Line graph in terms of its associated matrices: Adjacencies (L2LG), Costs (L2LC), and Next Nodes (L2LN). The latter is used for path finding, i.e. finding the actual path in terms of the nodes involved in every optimal path. The Graph Search component then uses these matrices in an All-Shortest-Path problem setting to find all Easiest Paths and the Travel Times associated with every pair of directed Origin-Destination nodes. A path-finding component is also provided to check individual Easiest Paths for arbitrary origin and destination pairs.

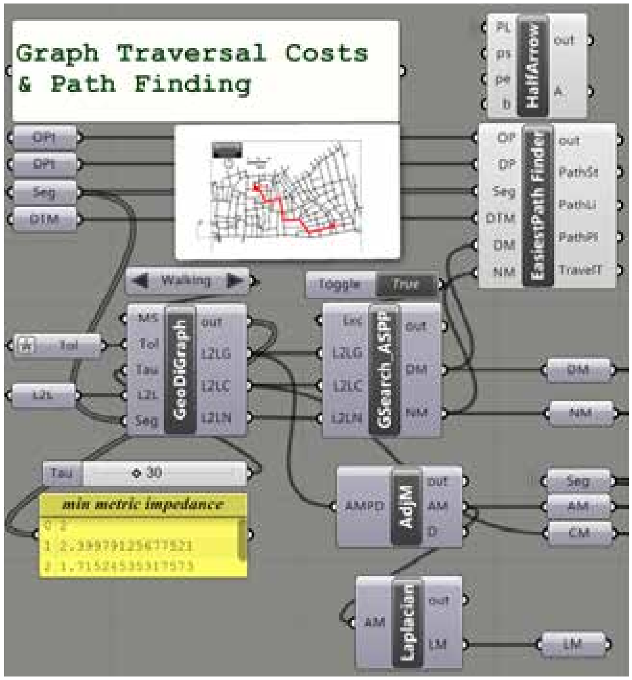


FIGURE 148 Graph construction and graph search components

§ 6.5.4 Polycentric Accessibility Measures (Figure 149)

This group of components works on a non-square matrix of distances called Relevant Distance Matrix (RDM) from/to a list of Points of Interests (POI), chosen by the user and then computes several accessibility measures on the network using these distances (To/From POI), namely: 1. Catchment (for All/Any POI), 2. Fuzzy Closeness (To All/To Any POI), 3. Inclusive Zoning (generalized Voronoi Diagram) (Any POI), 4. Exclusive Zoning (Any POI), and 5. ReachTree (i.e. Betweenness for All/Any POI). Each component (or method in the dll) computes a list of measures and a list of colours associated with it. These numbers and colours can then be attributed to either the streets (graph nodes) or the building plots. This latter attribution is based on a number of keys that relate each plot to its closest street. If better-detailed information is available then these keys can be overridden manually.

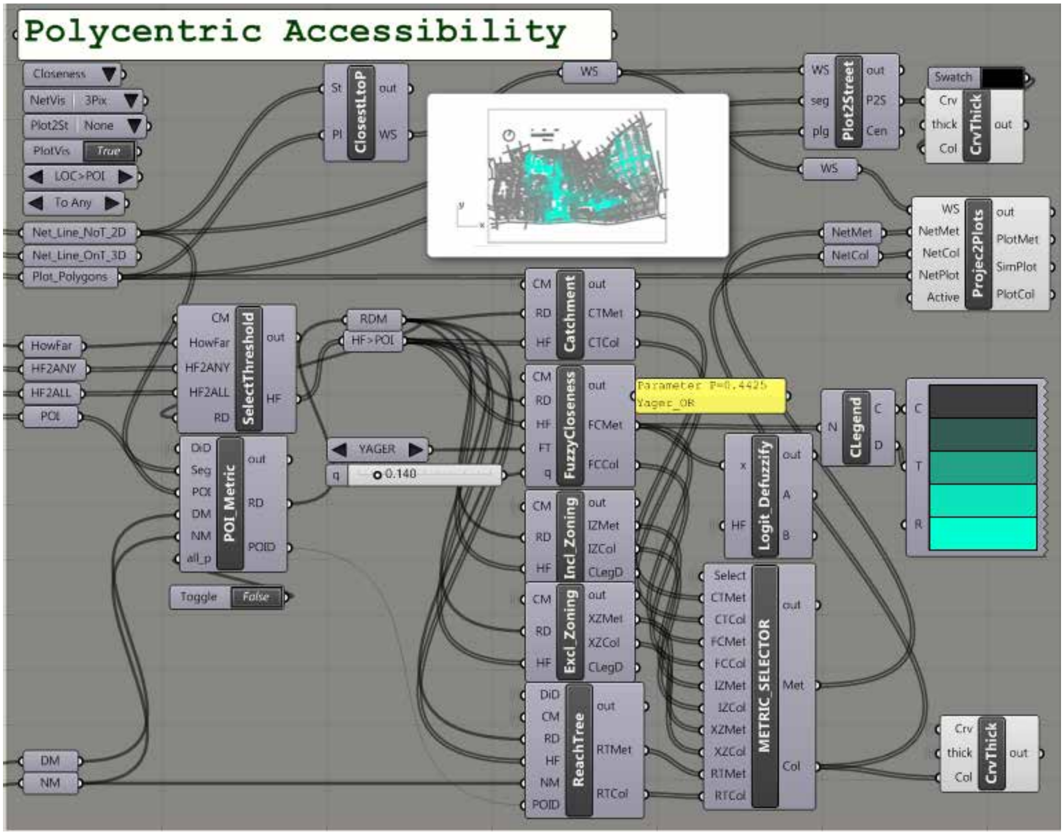


FIGURE 149 Polycentric Accessibility measures using distances to/from a list of POI

§ 6.5.5 Inputs for Polycentric Accessibility Analysis Tools (Figure 150)

The inputs of the accessibility analysis tools are Points of Interest as geometric points, and a corresponding list of travel time radii associated with them called How Far parameters. These are meant to allow the user to specify how far a person would be willing to travel on foot/by bike to/from POI.

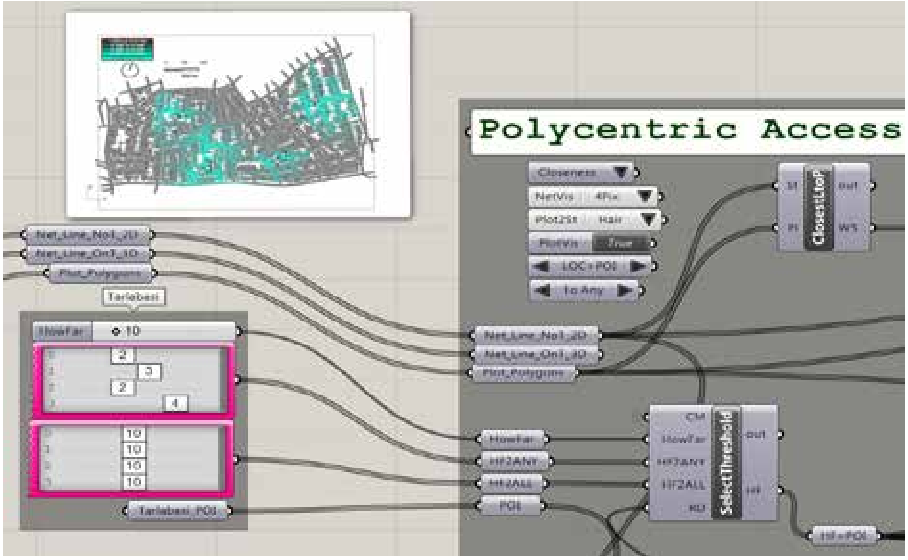


FIGURE 150 Inputs for Polycentric Accessibility analyses. It is more convenient to use two different sets of How Far parameters; because it might not reasonable to search for locations that are close to all POI considering small radii of travel time. If all locations are chosen as potential POI, then a single parameter for How Far is used.

§ 6.5.6 Geodesic Centrality Measures (Figure 151)

This group of tools directly use the outputs of graph search algorithm to find the importance of streets in terms of the likelihood of passage through them (betweenness as to Easiest Paths) and the likelihood of being common destinations (closeness to all locations via Easiest Paths). Betweenness computation is known to be very computationally demanding. In fact, the heaviest computational process in the whole toolkit is the betweenness centrality analysis. It is interesting, however, that using a local search, similar to the approach of Space Syntax, we can both lower the computational load and make more sense of analyses by considering only locations that possibly matter, i.e. those in a reasonable range of walking or cycling. Exactly because of this consideration, we can control the so-called edge effects by considering a margin -for a network that is to be studied- larger than the desired radius of search. For instance if we are interested in Betweenness of streets for walking trips of maximum 10 minutes long, then our network must have a margin that is bigger than 10 minutes times a typical walkable distance for a minute (1/60 *5KM=83.3 m).

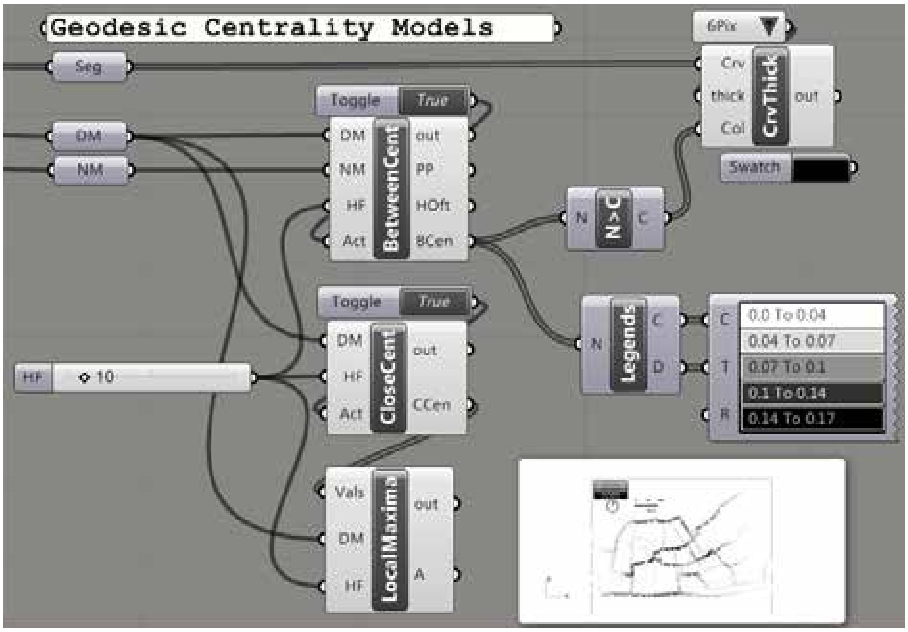


FIGURE 151 geodesic centrality tools, and a tool for finding local maxima using gradient descents.

§ 6.5.7 Spectral Centrality Measures (Figure 152)

We primarily implemented these tools for better understanding the spectral network analysis and as an intermediary steps towards realization of the Random Walk centrality analysis models. The tools currently are inefficient due to the inefficiency of the non-iterative solvers used from Math Net library for solving a system of linear equations. If the use of normalized beta parameter is not desired, then the user can use our generalized Power Iteration method after (Koren, 2003) to compute eigenvector centrality efficiently. This group of tools can be used in a number of different ways, which might not be equally interesting. These different ways of using the tools correspond to the different inputs possible for the tools: Adjacency Matrices, Weighted-Adjacency Matrices, or Distance Matrices. The latter has proven to be the most interesting case. The interesting connotation of using distance matrices as graphs is the fact that they literally translate the idea of the so-called first law of geography into a graph: "*Everything is related to everything else, but near things are more related than distant things*" (Tobler, 1970). The result of the supposedly abstract spectral analysis turns out to be very intuitive: it becomes comparable with closeness centrality. We have reported these methods in a paper in 7th SimAUD symposium[74].

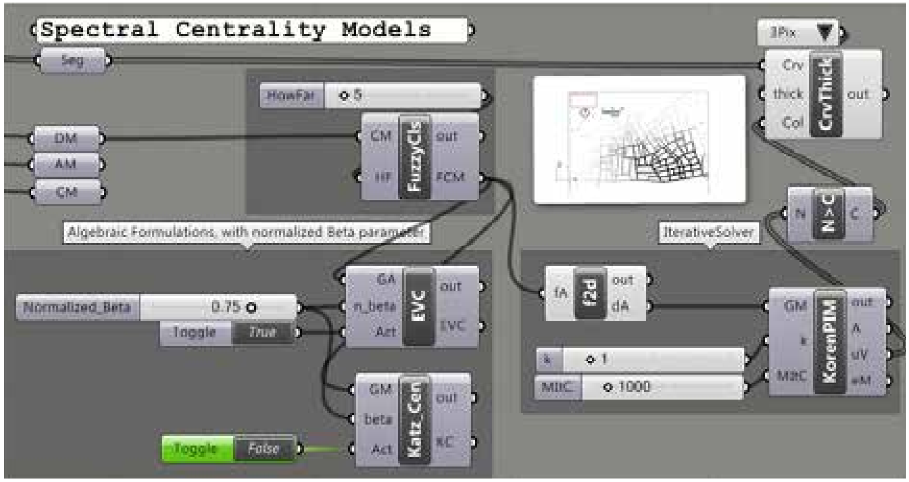


FIGURE 152 Spectral Centrality tools, including a state-of-the-art generalized Power-Iteration eigen-solver after (Koren, 2003)

§ 6.5.8 Random Walk Probabilistic Models (Figure 153)

This set of tools include our Random Walk models Alef, Bet, Gimel and Dalet from previous chapter for forming transition probability matrices and two generic solvers for finding the stationary probability distributions of random walks. One of these solvers is an algebraic solver based on the method explained in chapter 5 (implementing a non-iterative Simultaneous Linear Equation (SLE) solver from Math.Net); and the other one is an iterative solver using Power Iteration method.

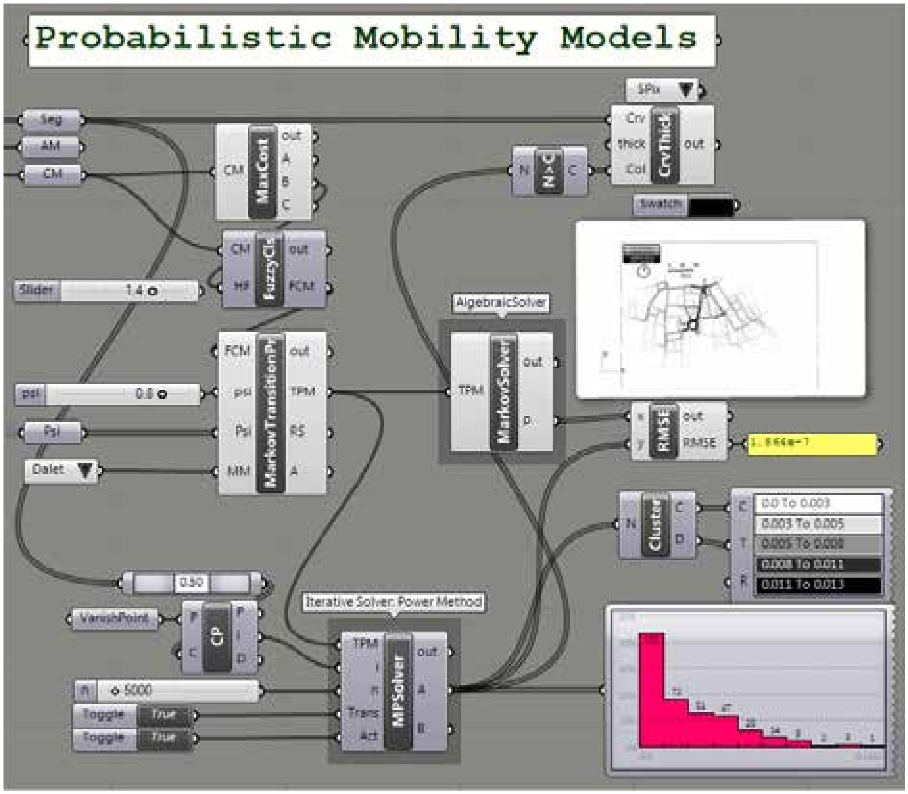


FIGURE 153 Random-Walk probabilistic models Alef, Bet, Gimel, and Dalet and two solvers for them. Models Gimel and Dalet require an external input called Psi (ψ probabilities of staying at streets).

§ 6.5.9 Spectral Graph Drawing Tools (Figure 154)

These tools implement the State-of-the-Art spectral Graph Drawing method of (Koren, 2003). This elegant and intuitive method helped very much in structuring the tools implementing methods from spectral graph theory. The Eigen-solver of Koren is a generalized Power Iteration method and therefore quite efficient for drawing large graphs. Implementing spectral drawing served two purposes: providing an intuitive view of spectral graph theory and paving the way for efficient eigenvector centrality analyses. It is also imaginable to use this method further for such purposes as clustering networks as to the distances of nodes in the lower dimensional Euclidean Space (i.e. in the topological embedding space).

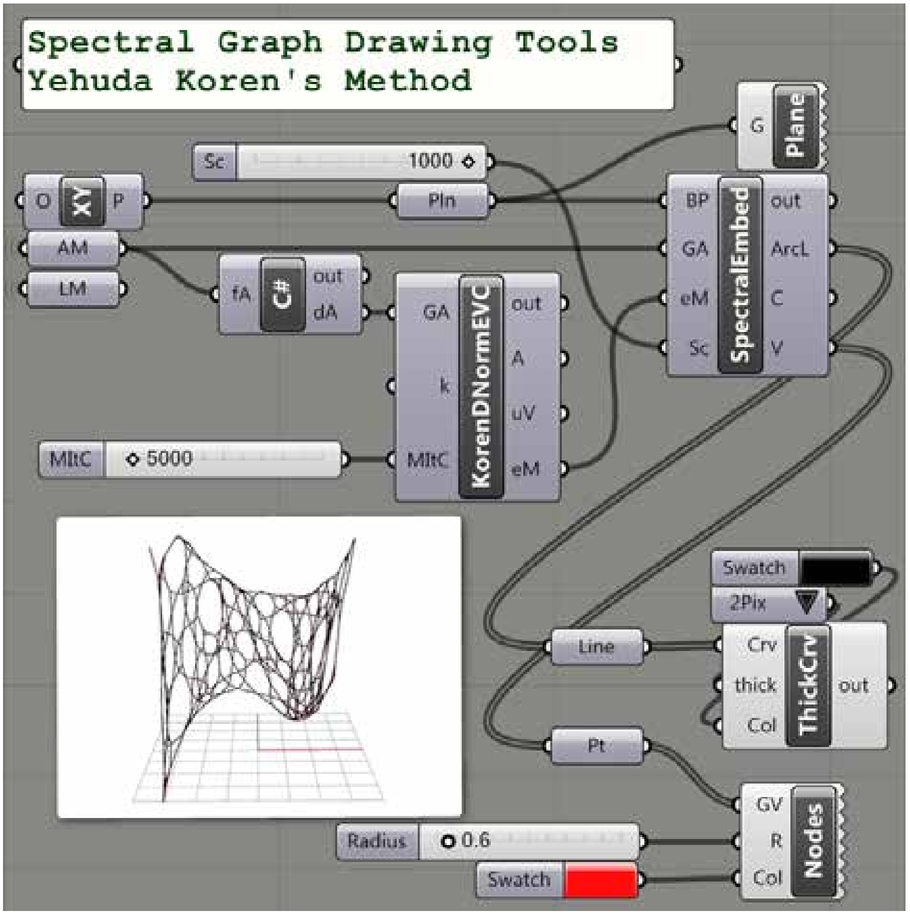


FIGURE 154 Spectral Graph Drawing Tools implementing the state-of-the-art method of (Koren, 2003)

§ 6.5.10 Matrix Plot Tools (Figure 155)

In our methodology, we advocate a distinction between topological graph theories from graph theory. We argue that graphs, as abstract relational models need not be represented geometrically in order to exist. We therefore provide tools to visualize graphs in their most abstract form as possible, i.e. a matrix plot view. This is a common practice in physics and mathematics that helps to dissociate the concept of a graph from what often comes to mind as a natural interpretation of a graph from street networks (i.e. a Point-to-Point/Junction-to-Junction adjacency graph). The idea is that the graph representing a spatial network does not need to resemble the shape of the network. In fact, by changing the way a graph is represented one can understand aspects that could be not easily observable in another representation.

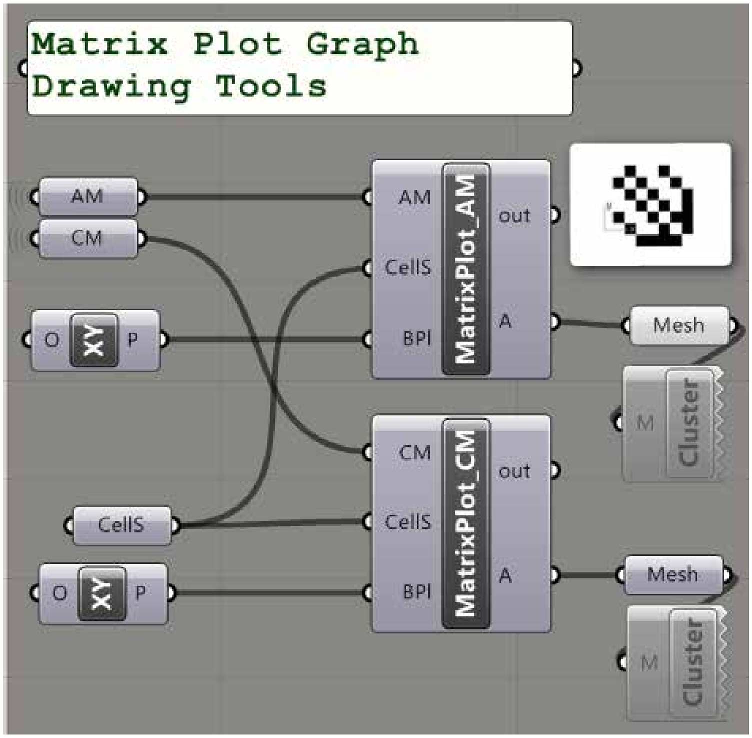


FIGURE 155 Matrix Plot graph drawing tools draw a pixelated image whose pixels correspond to the entries of graph matrices from top-left to bottom-right. If the graph is not weighted, i.e. only consisting of 0s and 1s, it will be drawn in black and white; otherwise, in case of weighted matrices colours will be attributed to indicate the weight (strength) of the connections, i.e. darker colours represent better connections (less traversal costs).

§ 6.5.11 Graph Drawing Utility Tools (Figure 156)

These tools are utilities necessary for drawing graphs using arcs or lines for representing edges (links). The principle is that the graph prior to embedding does not have any geometric representation. If a number of vertices are assigned to represent the nodes of the graph then connections between these vertices will be representative of links between the nodes in question. It is customary to show such links as arcs or straight lines. As the input graphs might be directed (in case there is a topographic terrain) then arrows might ne necessary to indicate the direction of arc-links or edge-links.

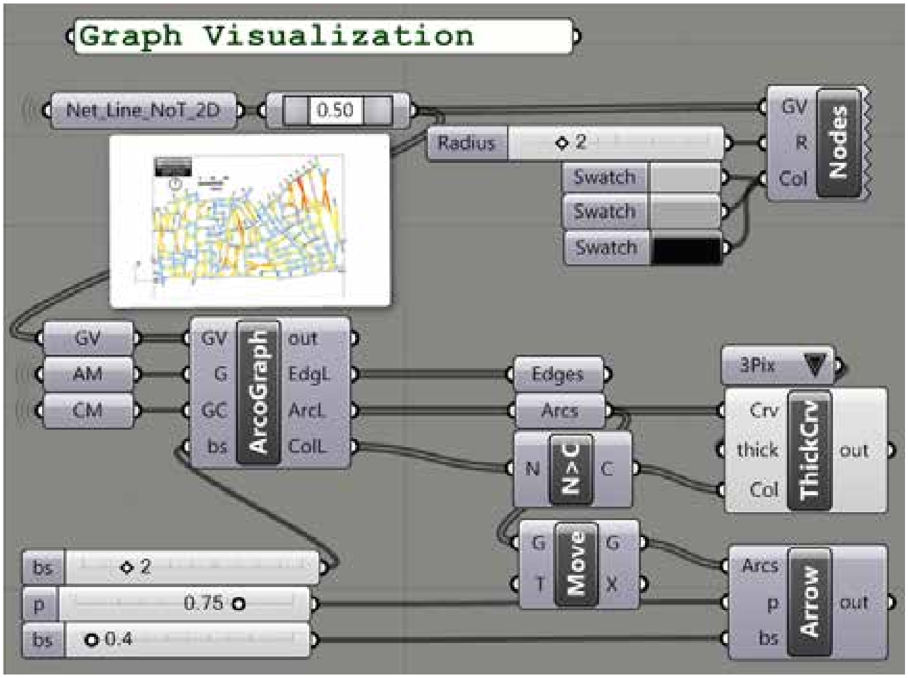


FIGURE 156 Utility tools for drawing spatial network graphs

§ 6.5.12 Quantitative Validation & Calibration Tools (Figure 157)

We have envisioned that quantitative validation and calibration workflows will be necessary in future developments of the tool. To this end, we have included a number of utility tools for implementing "Regression Validation" of hypotheses. These tools provide an open interface to Math.Net.Numerics.Statistics namespace to access methods such as Pearson Correlation Coefficient, Regression Line, Covariance and alike. In addition, parametric diagram drawing tools, a Root Mean Square Error indicator, and a calculator for Shannon's Entropy (of information content) with three units of *Bits* (Basic Information Units), *Nits* (Natural Information Units) and *Dits* (Hartley's Decimal Digit Information Units) are provided as utilities.

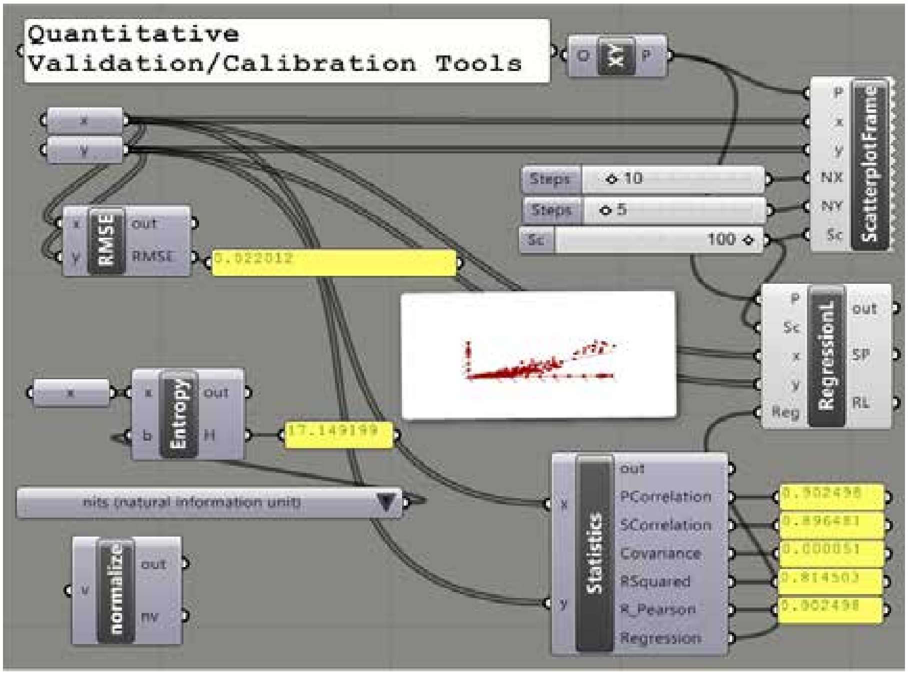


FIGURE 157 Utility Tools for Quantitative Validation and Calibration Workflows, Statistical Methods are provided openly from Math.Net.Statistics library

§ 6.6 Implementation Details and Issues

The act of designing and implementing the methods introduced in the previous chapter has been iterative. This is because each implementation of the methodology has been tested and revised accordingly. The data structures, algorithms, interfaces (inputs and outputs of functions), dependencies (on external libraries) have changed radically at least three times. The most subtle issues pertain to construction of a valid topological data model of a space network (that is also suitable from a cognitive perspective) and then representation of graphs, graph algorithms and eigen systems.

§ 6.6.1 Network Topological Models

The street-network data-models are often not directly usable for network studies because they usually require some cleaning, simplification, generalization, and validation prior to constructing graph models. In other words, we first need to make a valid topological model from a bunch of polylines in order to represent connectivity between street segments. To this end, one can follow two different approaches, each of which demanding an entirely different set of tools. We have already shown a set of 'simple' tools in sections 6.5.1 and 6.5.2 for this purpose, here we show a more systematic approach for dealing with networks from resources such as OpenStreetMap.

§ 6.6.1.1 Vector-Based Approach

This approach is a systematic application of basic methods previously shown in sections 6.5.1 and 6.5.2. First, a set of polylines representing streets are intersected with one another to split them at junctions. Then, the polylines are simplified between junctions. The long segments of the simplified polylines then will be shattered into pieces maximally as long as the desired resolution specified by the user. Note that this metric value acts like a 'level of detail' for the study and is essentially different from the 'tolerance for error'. The polylines then will be reduced to lists of line segments while removing short segments (by projecting them to nearby vertices). These steps are shown in Figure 158.

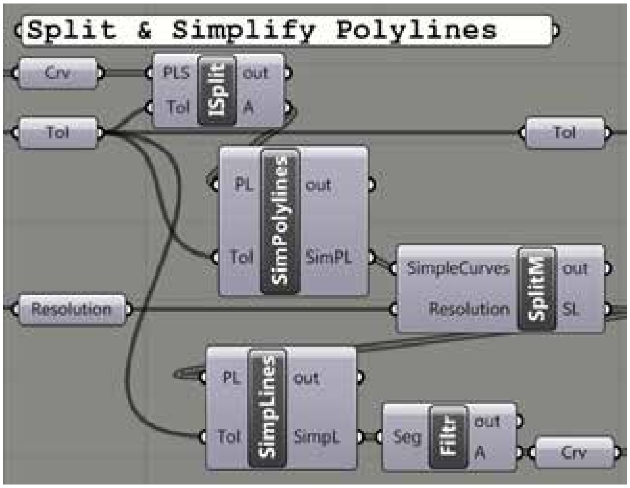


FIGURE 158 tools for splitting and simplifying street polylines

Using the same tolerance and the line segments generated so far, we make a temporary Bipartite Adjacency model in order to check whether the line segments and their junctions represent 'connected graphs'. For this purpose we use a Connected Components algorithm based on recursive DFS (Depth First Search) and take the largest relevant connected component to continue. Using this connected component, we redraw street segments as edges of this graph. Using these tools, the user can see if there are islands in the network.

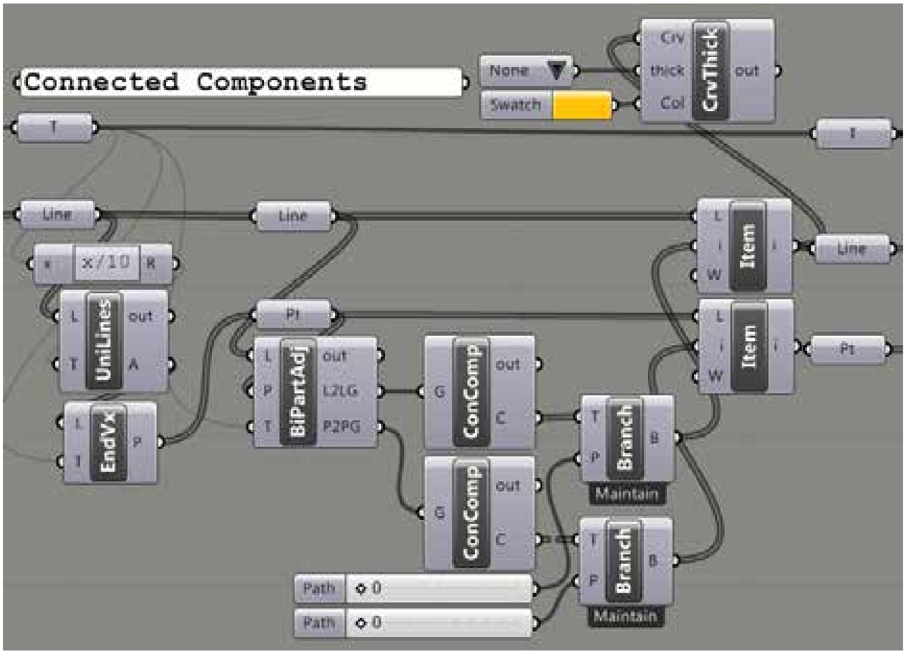


FIGURE 159 tools for reconstructing the largest relevant connected component of a spatial network

The next step in determining which streets are topologically connected (adjacent to) to which other streets is performed by constructing a list of vertex points and a list of edge lines. Then we construct a bipartite graph to model how these points and lines are adjacent to each other. We then construct the edges of the $\boldsymbol{A}^{VV}$ adjacency graph and use them as our street segments (Figure 160). Finally (Figure 161), we construct our $\boldsymbol{A}^{EE}$ graph using these edges.

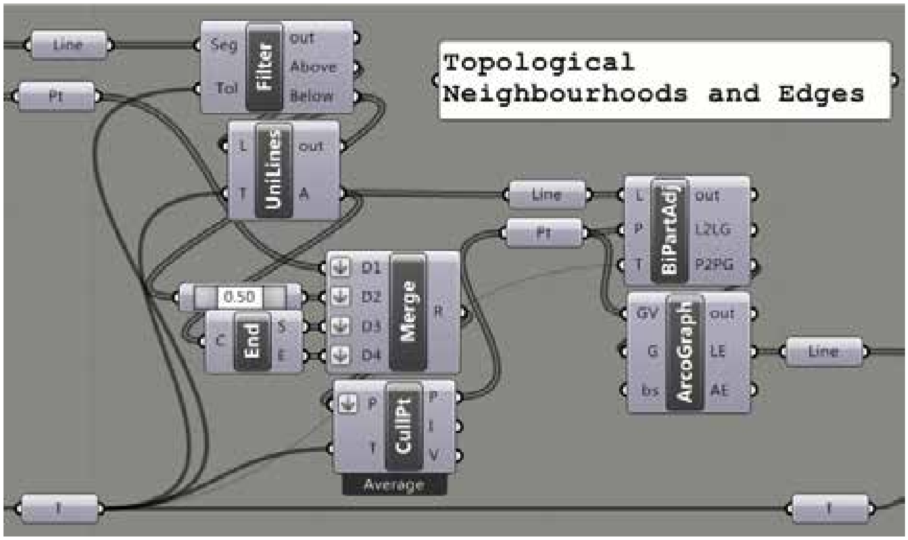


FIGURE 160 redrawing the network as edges of the Point-to-Point graph

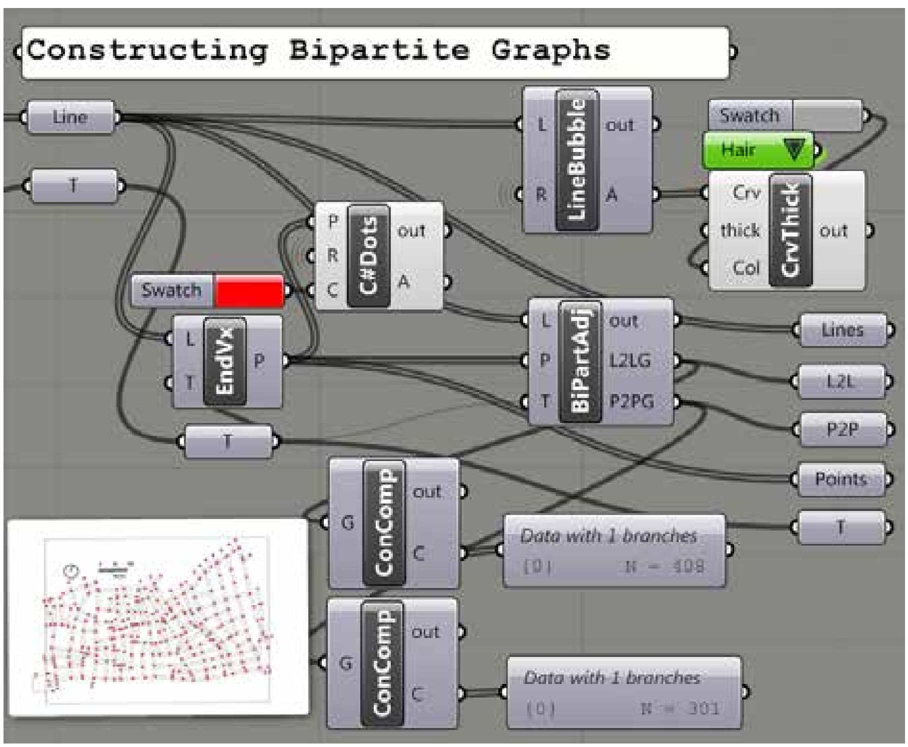


FIGURE 161 constructing the Line-to-Line adjacency graph from undirected network edges

§ 6.6.1.2 Raster-Based Approach

This approach is based on pixelating or voxelating the street network topologically at 4-connectivity or 8-connectivity levels (Laine, 2013) and (Nourian, P, Goncalves, R, Zlatanova, S, Arroyo Ahori, K, Vo, A, 2016) to obtain a raster model of the network as a binary image. The concepts of 4-connectivity or 8-connectivity correspond to 4-neighbourhoods (a.k.a. von Newman neighbourhoods[a]) and 8-neighbourhoods (a.k.a. Moore neighbourhoods[b]). This approach is flexible enough to accept streets as polygonal models as well. This flexibility could simplify using and combining alternative data models such as those of European Road Atlas datasets, which are available as road polygons. It could even be that a binary image as such be generated directly from satellite imagery using image-processing techniques. In any case, a binary image as such can be later processed to find the connected components; and 'thinned' using techniques from Mathematical Morphology so as to obtain a Topological Skeleton of the road space network. A common algorithm for this purpose is the Zhang-Suen thinning algorithm (Zhang, T. Y., and Ching Y. Suen., 1984). Once the skeleton is available, we can vectorise it to obtain a valid street network. We have developed and implemented the raster steps of this approach because of its promising flexibility and scalability; however, the work is yet in progress.

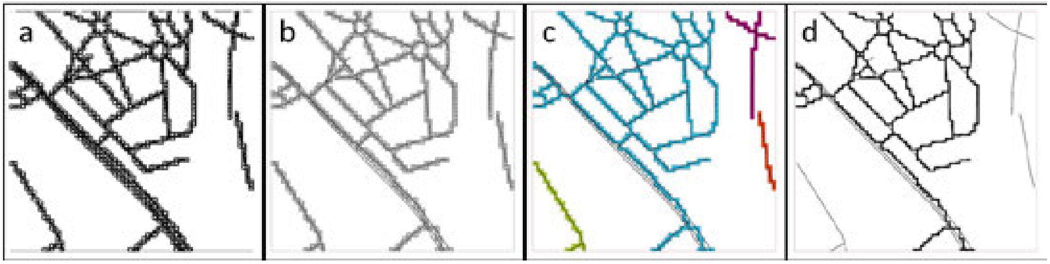


FIGURE 162 shows basic steps of a raster-based approach for building a valid network topological model: a) topological pixilation, b) morphological thinning, c) connected components, d) network reconstruction

a Weisstein, Eric W. "von Neumann Neighborhood." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/vonNeumannNeighborhood.html

b Weisstein, Eric W. "Moore Neighborhood." From MathWorld--A Wolfram Web Resource. http://mathworld.wolfram.com/MooreNeighborhood.html

§ 6.6.2 Graph Data Models

We started representing undirected graphs as adjacency lists, first in a dictionary-like format particular to the Grasshopper3D environment to ensure easy readability of data flows. However, due to the extremely high [computational] cost of retrieval of items (links) from this data structure. We then switched to an alternative structure as $List < List < int >>$ for representing adjacency lists and optimized it further to an array of lists as $List < int > []$. Until this point, we were using the Dijkstra algorithms for finding optimal paths. However, as we shifted our focus on the Spectral Graph Theory algorithms, we found Floyd-Warshall algorithm a more suitable choice as it naturally works with a matrix representation of graphs. In other words, as we needed to work with graphs in matrix form, we realized it would be more straightforward to use the Floyd-Warshall algorithm, which processes a matrix of distances and succeeding nodes. With this implementation, that is in the latest version of configraphics.dll and CONFIGURBANIST, graphs are initially modelled as three 'sparse matrices' (fist as array of arrays, a.k.a. jagged arrays and then using the Math.NET library), namely an adjacency matrix, cost matrix and succeeding matrix. This data model comes with a number of advantages and disadvantages.

The advantages of a Matrix-Based data model for graphs are:

- Ease of running Linear Algebraic algorithms (e.g. those of Math.NET library);
- Good readability of the graphs, and clarity of the representation as disassociated from geometric or topologic connotations (points, lines, vertexes and edges are no longer present in the graph data model) and so the generality of the graph data structure is at peak; and
- Clear distinction between un-weighted adjacency graph and the weighted adjacency graph (cost matrix) ensuring ease of different operations on them

The disadvantages of a Matrix-Based data model for graphs are:

- Large memory footprint even when sparse matrices are used (compared to adjacency lists);
- Using Dijkstra algorithm would be rather inconvenient/complicated with graph matrices; and
- The overhead of using an external library for matrix classes and operations

§ 6.6.3 Graph Search Algorithms

We need to explain two subtle points about the abovementioned disadvantages: Dijkstra algorithm is often reported to have a theoretical time complexity of $O(|E| + |V| \log |V|)$. This can be misleading as this asymptotic time complexity could be achieved only when a priority queue is implemented for the search of minimum-weight edge in the search. Otherwise, i.e. with a naïve implementation using a list of array of edges, one must simply spend time $O(n)$ for the search for the minimum-cost edge. This means that in that case the running time would be of $O(|E| + |V||V|)$, which in case of sparse graphs would be equal to $O(|V|^2)$. This is because for a complete graph $|E| = |V|^2$, but in case of sparse graphs $|E| \ll |V|^2$; therefore the running time will be determined by the number of vertices. Noting that the Dijkstra algorithm is a one-to-many shortest path algorithm we can see that we would require running this algorithm $n = |V|$ times to achieve what Floyd-Warshall algorithm does. That would mean with a simple Dijkstra implementation our theretical time complexity for a all-pairs shortest path problem would be $O(|V|^3) = O(n^3)$. Now consider that Floyd-Warshall algorithm also has a time complexity of $O(n^3)$. However, Flowd-Warshall has a much simpler structure and fewer and simpler arithmetic operations so it will be better in practice and also for our case, compared to running single-source shortest path algorithm (Dijkstra) for n times; note that we also have to produce distance matrices anyway.

In addition to this point, we opted for simplicity and clarity in our code, because a simple, slightly less efficient maintainable code would be preferable to an over-engineered complicated code. From an engineering point of view, if the problem at hand is very well formulated one must naturally opt for the most efficient way of implementation. However, this view occludes the fact that the very definition of the problem at hand is ill structured and subject to many cycles of "design, implementation, and test". Therefore, we can justify our choice of data-structures from a rather qualitative point of view, say regarding efficacy, but not efficiency[75].

§ 6.6.4 Linear Algebraic Algorithms

After our last rounds of test and implementation, we realized that in fact we do not need much functionality from a library such as Math.NET, because even for the most sophisticated algorithms such as finding eigenvalues and eigenvectors we have ended up developing our own solutions. The remainder of operations that are still dependant on Math.NET are simple matric multiplications and solvers for systems of linear equations that are used in computing generalized Eigenvector Centrality and

Katz Centrality. Although we are using the best option we could find for computing these two centrality indicators (that is solving a system of linear equations instead of inverting a matrix) their computations times are still the worst among all other indicators, with the exception of betweenness centrality that is in any case notorious for its extreme computational cost.

These solutions are algebraically elegant but computationally still prohibitively inefficient for large networks. They can be possibly improved by using iterative SLE solvers from Math[76]. However, we decided to discontinue working on these generalized eigenvector centrality measures in favour of our own generalized formulation of eigenvector centrality measure that is computed using our Power Iteration method and a Fuzzy Graph Model as reported in our forthcoming paper in SimAUD 2016: "Spectral Graph Spectral Modelling of Spatial Networks".

§ 6.6.4.1 Eigenvector Centrality

Our formulations of the eigenvector centrality ranking index (Bonacich's formulation) here:

$$implicit: \boxed{(\boldsymbol{I} - \beta\boldsymbol{A})\boldsymbol{c}^e = \alpha\boldsymbol{A}\boldsymbol{e}} \qquad explicit: \boxed{\boldsymbol{c}^e = \alpha(\boldsymbol{I} - \beta\boldsymbol{A})^{-1}\boldsymbol{A}\boldsymbol{e}}$$

Algorithm 12: Eigenvector Centrality, Bonacich's formulation (Bonacich, 1987)

```
Inputs: Marix[n×n] A, double n_beta//for Adjacency Matrices, using normalized beta (see Chapter 5)
Outputs: Vector[] B_EVC //arrays of results Bonacich's EVC
vector[n] B_EVC=new vector[n];//Eigenvector Centrality Indexes will be stored in this vector
Matrix[n×n ] I=new Matrix[n×n].Identity();
Matrix[n×n] B=A + I;//ensuring convergence for finding Perron-Frobenius eigenvalue or spectral radius
vector[n] DEVC= new vector[n];//disposable dominant eigenvector
vector[n] NEVC= new vector[n];//disposable dominant eigenvector
vector[n] Error= new vector[n];//deviation between current and previously computed EVC
do{
o    NDEVC = DEVC;
o    NDEVC = B * DEVC;
o    NDEVC = DEVC.Normalize(2);
o    Error = NDEVC - DEVC;
o    counter += 1;
}while(Error.Norm(2) > 0.001 && counter < 10E03)
double r = (B * DEVC).Norm(2) / DEVC.Norm(2);//Perron-Frobenius eigenvalue, aka spectral radius using Rayleigh Quotient
double beta = n_beta * (1 / r);//refer to chapter 5 for an explanation of this tweak
Matrix[n×n] C=I-beta*A;
vector[n] e= new vector[n,1];//identity vector
vector[n]d= A*e;//effectively holding node-degree values
B_EVC=C.Solve(d);//using an iterative solver for the system of linear equations in the form Ax = b
```

We hereby provide the source for these two centrality indices[77].

SourceCode 2: Eigenvector Centrality, Bonacich's formulation (Bonacich, 1987) in C#, using Math.NET

```
 private void RunScript(object GA, double n_beta, bool Act, ref object EVC)
        {
            if (!Act) { return; }
            MathNet.Numerics.LinearAlgebra.Matrix<float> AM =
(MathNet.Numerics.LinearAlgebra.Matrix<float>)GA;
            int N = AM.ColumnCount;

            MathNet.Numerics.LinearAlgebra.Matrix<float> B = AM +
MathNet.Numerics.LinearAlgebra.Single.SparseMatrix.CreateIdentity(N);
            MathNet.Numerics.LinearAlgebra.Vector<float> DEVC =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
            MathNet.Numerics.LinearAlgebra.Vector<float> NDEVC =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
            //
            MathNet.Numerics.LinearAlgebra.Vector<float> Error =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
            int counter = 0;
            do
            {
                NDEVC = DEVC;
                NDEVC = B * DEVC;
                NDEVC = DEVC.Normalize(2);
                Error = NDEVC - DEVC;
                counter += 1;
            } while (Error.Norm(2) > 0.001 && counter < 10E03);
            double r = (B * DEVC).Norm(2) / DEVC.Norm(2);//Perron-Frobenius
eigenvalue, aka spectral radius
            double beta = n_beta * (1 / r);//refer to chapter 5 for an
explanation of this tweak
            Print(String.Format("Spectral Radius:{0}", r.ToString()));

            float betaF = (float)beta;
            MathNet.Numerics.LinearAlgebra.MatrixBuilder<float> IB =
MathNet.Numerics.LinearAlgebra.Matrix<float>.Build;
            MathNet.Numerics.LinearAlgebra.VectorBuilder<float> CB =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build;
            MathNet.Numerics.LinearAlgebra.VectorBuilder<float> eB =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build;

            MathNet.Numerics.LinearAlgebra.Matrix<float> I =
IB.DiagonalIdentity(N);
            MathNet.Numerics.LinearAlgebra.Vector<float> C = CB.Dense(N);
            MathNet.Numerics.LinearAlgebra.Vector<float> e = eB.Dense(N, 1);
            MathNet.Numerics.LinearAlgebra.Vector<float> b = AM * e;//the
solution for the system of linear equations

            MathNet.Numerics.LinearAlgebra.Matrix<float> I_betaR = I - betaF
* AM;
            C = I_betaR.Solve(b);
            EVC = C;

        }
```

§ 6.6.4.2 Katz Centrality:

We recite our formulations of the Katz centrality ranking here:

$$implicit: \boxed{(\boldsymbol{I} - \beta\boldsymbol{A})\boldsymbol{c}^{\boldsymbol{Katz}} = \beta\boldsymbol{A}\boldsymbol{e}} \quad explicit: \boxed{\boldsymbol{c}^{\boldsymbol{Katz}} = \beta((\boldsymbol{I} - \beta\boldsymbol{A})^{-1}\boldsymbol{A})\boldsymbol{e}}$$

Algorithm 13: Katz Centrality

Inputs: Marix[$n\times n$] A, **double** n_beta//for Adjacency Matrices, using normalized beta (see Chapter 5)
Outputs: Vector[] K_EVC //arrays of results Katz's EVC

vector[n] K_EVC=new **vector[n]**;//Eigenvector Centrality Indexes will be stored in this vector
Matrix[n×n] I=new **Matrix[n×n]**.Identity();
Matrix[n×n] B=A + I;//ensuring convergence for finding Perron-Frobenius eigenvalue or spectral radius
vector[n] DEVC= new **vector[n]**;//disposable dominant eigenvector
vector[n] NEVC= new **vector[n]**;//disposable dominant eigenvector
vector[n] Error= new **vector[n]**;//deviation between current and previously computed EVC
do{//Power Iteration
○ **NDEVC = DEVC;**
○ **NDEVC = B * DEVC;**
○ **NDEVC = DEVC**.Normalize(2);
○ **Error = NDEVC - DEVC;**
○ counter += 1;
}**while**(**Error**.Norm(2) > 0.001 && counter < 10E03)
double r = (**B** * **DEVC**).Norm(2) / DEVC.Norm(2);//Perron-Frobenius eigenvalue, aka spectral radius using Rayleigh Quotient
double beta = n_beta * (1 / r);//refer to chapter 5 for an explanation of this tweak
Matrix[n×n] C=I-beta****A;**
vector[n] e= new **vector[n,1]**;//identity vector
vector[n] bd= beta***A*****e**;//effectively holding node-degree values
K_EVC=**C**.Solve(**bd**);//using an iterative solver for the system of linear equations in the form $\boldsymbol{Ax} = \boldsymbol{b}$

Note that instead of inverting the matrix, as in the explicit formulation of both eigenvector centrality rankings we choose to solve the implicit equation as it involves solving a system of linear equations. This is because inverting matrices is a very computationally demanding procedure. In practice, it is best to avoid such inversions wherever possible. However, a non-optimized method for solving a system of linear equations can be as costly as inverting a matrix, that is $O(n^3)$. This issue seems to be the case with the method we have used from the Math.NET library. However, as said before, this method can be replaced by iterative solvers to improve the performance significantly, e.g. by using iterative Krylov subspace methods to $O(n)$.

SourceCode 3: Katz Centrality (Katz, 1953), using Math.NET

```
private void RunScript(object GM, double beta, bool Act, ref object KC)
  {
    if (!Act){return;}
    MathNet.Numerics.LinearAlgebra.Matrix<float> AM =
(MathNet.Numerics.LinearAlgebra.Matrix<float>) GM;
    int N = AM.ColumnCount;
    MathNet.Numerics.LinearAlgebra.Matrix<float> B = AM +
MathNet.Numerics.LinearAlgebra.Single.SparseMatrix.CreateIdentity(N);
    MathNet.Numerics.LinearAlgebra.Vector<float> DEVC =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
    MathNet.Numerics.LinearAlgebra.Vector<float> NDEVC =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
    //
    MathNet.Numerics.LinearAlgebra.Vector<float> Error =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
    int counter = 0;
    do
    {
      NDEVC = DEVC;
      NDEVC = B * DEVC;
      NDEVC = ((float) (1 / DEVC.Norm(2))) * DEVC;
      Error = NDEVC - DEVC;
      counter += 1;
    }while(Error.Norm(2) > 0.0001 && counter < 10E03);
    double r=(B * DEVC).Norm(2) / DEVC.Norm(2);//Perron-Frobenius eigenvalue,
aka spectral radius

    Print(String.Format("Spectral Radius:{0}", r.ToString()));

    float b = (float) (beta * (1 / r));
    MathNet.Numerics.LinearAlgebra.Matrix<float> betaA = b * AM;
    MathNet.Numerics.LinearAlgebra.Single.SparseMatrix I =
MathNet.Numerics.LinearAlgebra.Single.SparseMatrix.CreateIdentity(N);
    MathNet.Numerics.LinearAlgebra.Matrix<float> I_betaA = I - betaA;
    MathNet.Numerics.LinearAlgebra.Vector<float> e =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
    MathNet.Numerics.LinearAlgebra.Vector<float> cKatz =
MathNet.Numerics.LinearAlgebra.Vector<float>.Build.Dense(N, 1);
    MathNet.Numerics.LinearAlgebra.Vector<float> betaAe = betaA * e;
    cKatz = I_betaA.Solve(betaAe);
    KC = cKatz;


  }
```

§ 6.6.4.3 Few Dominant Eigenvectors

This is a very efficient and intuitive 'power iteration method' that we have generalized from a spectral graph drawing method (Koren, 2003). It is for finding the first few dominant eigenvectors of a graph matrix, the single eigenvector version of this algorithm is widely used in solving Eigenvector centrality (the simple formulation such as that of accessibility by (Gould, 1967)) Markov Chain/Random Walk problems, Google Page Rank. This generalized version can be used in Spectral Graph Drawing, Principal Component Analysis, Graph Clustering, and similar applications that need the first few dominant eigenvectors.

Algorithm 14: Compute First Dominant Eigenvectors, after Yehuda Koren (Koren, 2003)

Inputs: Marix[$n\times n$] M, **int** k, **int** MaxIter//for Hermitian Matrices
Outputs: Vector[] EVecs, **double**[]EVals //arrays of results

vector[n][k] $\boldsymbol{u}$=new **vector[n]**[k];//EVecs
double[k] lambda=**new double**[k];//Evals
for (int i = 0;i < k;i++){
▪ **double** CoDir=0; **int** counter=0;
▪ **vector** $\widehat{\boldsymbol{u}}_i$=Vector.Random(n); $\widehat{\boldsymbol{u}}_i$.Normalize(2);//p-norm
▪ do{
  ○ $\boldsymbol{u}[i] = \widehat{\boldsymbol{u}}_i$;
  ○ **for (int j = 0;j <i;j++) {**
    • $\boldsymbol{u}[i] = \boldsymbol{u}[i]-\left(\frac{\langle u[i],u[j]\rangle}{\langle u[j],u[j]\rangle}\right)*\boldsymbol{u}[j]$;//orthogonalize against previous eigenvectors
    • **u[i].Normalize(2);**
  ○ }
  ○ $\widehat{\boldsymbol{u}}_i = \boldsymbol{M}*\boldsymbol{u}[i]$; $\widehat{\boldsymbol{u}}_i$.Normalize(2);
  ○ CoDir = $\langle \widehat{\boldsymbol{u}}_i, \boldsymbol{u}[i]\rangle$; counter+=1;
▪ }**while**((CoDir<$1-\varepsilon$) ∩ (counter<MaxIter))
▪ $\boldsymbol{u}[i] = \widehat{\boldsymbol{u}}_i$;
▪ }
 	
for (int i = 0;i < k;i++){ lambda[i]= $\langle \boldsymbol{u}[i], \boldsymbol{M}*\boldsymbol{u}[i]\rangle$**;}**
EVecs= $\boldsymbol{u}$**; EVals=lambda;**

Mathematically, we have explained why this method works in chapter 5, in explaining eigenvector centrality. Here we give a simple verbal explanation. Every vector, e.g. a random vector in the Hilbert space ($x \in \mathbb{R}^n$) can be represented as a linear combination of eigenvectors of a Self-Adjoint operator (e.g. a Symmetric Graph Matrix).

§ 6.7 The Architecture of CONFIGRAPHICS.DLL

The current architecture of CONFIGRPHICS library only includes the methods of CONFIGURBANIST toolkit; however, it is envisioned to include those of SYNTACTIC in future. The library is made to make the whole application more easily accessible and adaptable for versatile workflows. Our strategy is to offer an open API (Application Programming Interface) that is not necessarily limited to a platform such as Rhino & Grasshopper 3D. At present, however, there are still dependencies on Rhinocommon.dll, mainly in geometric operations and type definitions. This version of the library is a collection of static C# (equal to shared methods in VB) methods[78], structured in four namespaces named Geometry, Topology, Graph Theory, and Spectral Graph Theory. The main namespace is call Spatial Network Analysis.

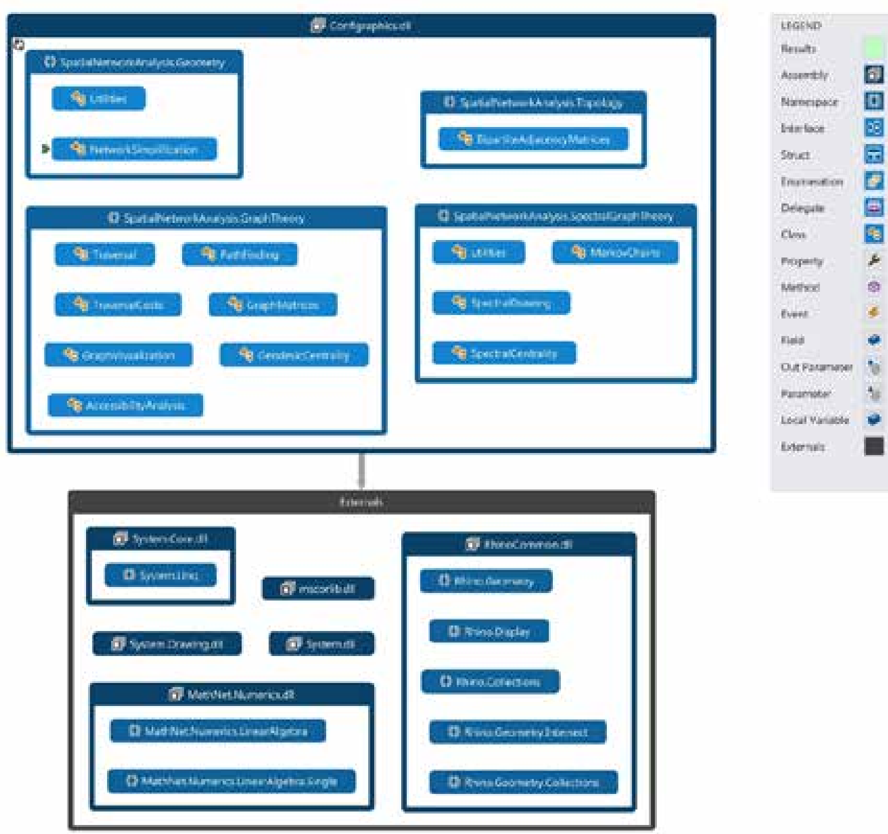


FIGURE 163 the latest version of configraphics library, its namesapeces, classes, and its dependencies

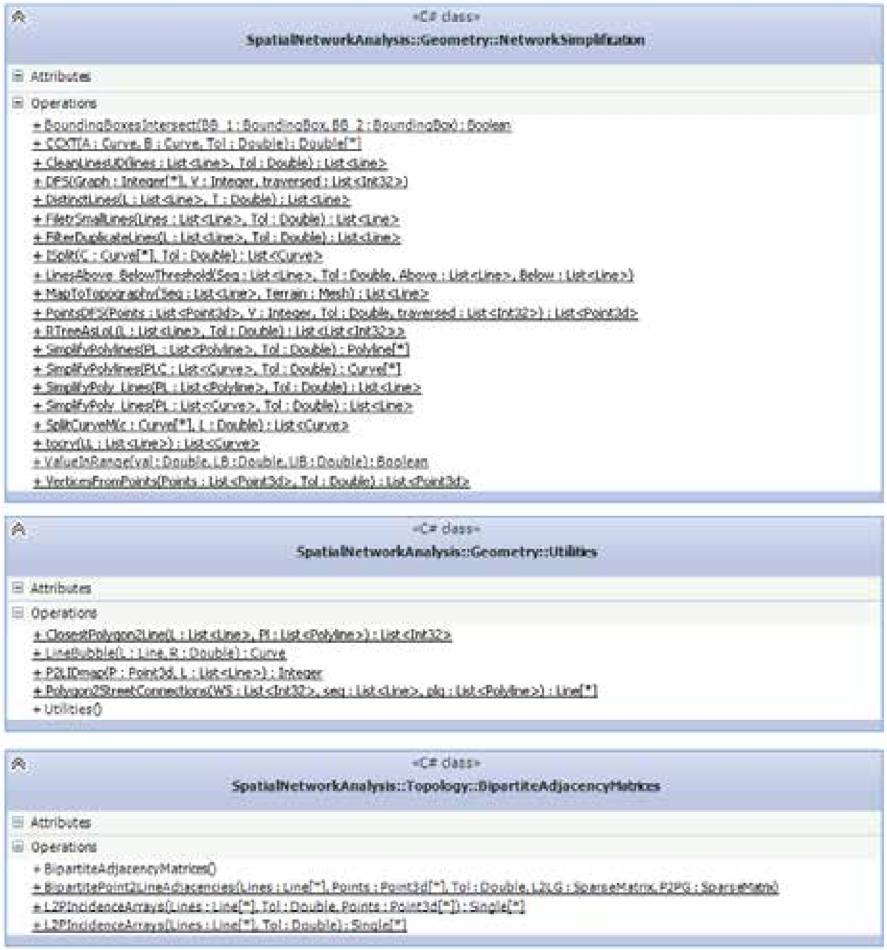


FIGURE 164 the methods in Geometry and Topology namespaces of configraphics

An example code written using the “Geometry” namespaces can be seen as a replacement of the modules previously shown in Figure 158 (Split & Simplify Network):

SourceCode 4: Split & Simplify a Street Network using configraphics.dll

```
var PL =
SpatialNetworkAnalysis.Geometry.NetworkSimplification.ISplit(PLS.ToArray(),To
l);
var SCL =
SpatialNetworkAnalysis.Geometry.NetworkSimplification.SimplifyPolylines(PL,To
l);
var LS =
SpatialNetworkAnalysis.Geometry.NetworkSimplification.SplitCurveM(SCL,Res);
var S =
SpatialNetworkAnalysis.Geometry.NetworkSimplification.SimplifyPoly_Lines(LS,
Tol);
var SimpleLines =
SpatialNetworkAnalysis.Geometry.Utilities.FiletrSmallLines(S, 0);
```

FIGURE 165 the methods in the Graph theory namespace of configraphics

An exemplary use of the “Graph Theory” namespace of the library is shown below:

SourceCode 5: using configraphics.dll for computing all Easiest Paths and distances corresponding to them

```
var L2LCM = (MathNet.Numerics.LinearAlgebra.Single.SparseMatrix) L2LC;
var L2LNM = (MathNet.Numerics.LinearAlgebra.Single.SparseMatrix) L2LN;
float[][] Costs = L2LCM.ToRowArrays();
float[][] Nexts = L2LNM.ToRowArrays();
SpatialNetworkAnalysis.GraphTheory.PathFinding.FindAllGeodesics(ref Costs,
ref Nexts);
DM = MathNet.Numerics.LinearAlgebra.Single.SparseMatrix.OfRowArrays(Costs);
NM = MathNet.Numerics.LinearAlgebra.Single.SparseMatrix.OfRowArrays(Nexts);
```

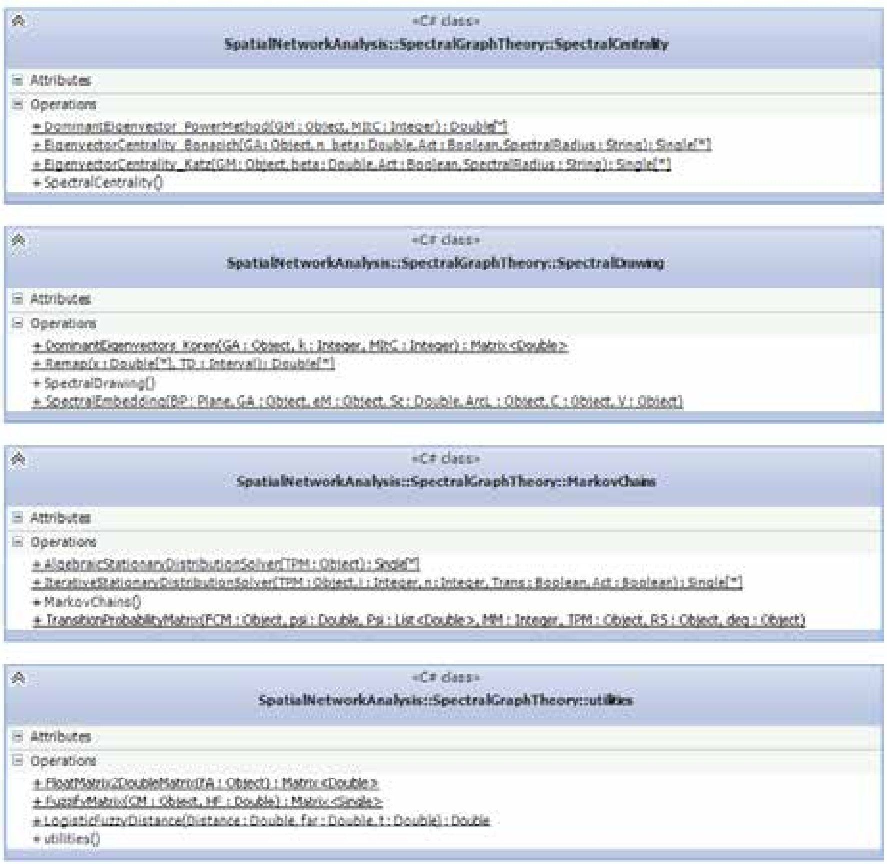


FIGURE 166 the methods in the Spectral Graph Theory namesapce of configraphics

SourceCode 6: using configraphics.dll for computing eigenvector centrality of a Fuzzy Grap

```
float[]EVC =SpatialNetworkAnalysis.SpectralGraphTheory.SpectralCentrality.
DominantEigenvector_PowerMethod(FGM, MI);
```

§ 6.8 Generating Accessibility Evaluation Reports

For comparing development plans or planning scenarios in terms of walking or cycling accessibility of amenities, we can generate accessibility evaluation reports to compare aggregate numerical values per plan or scenario to be able to assess their aptitude objectively in comparison to one another. A simple example could be thought of as a comparison of the percentage of residential plots having a good, e.g. 5 minutes walking access to any grocery store and a 10 minutes access to all of locations such as a train station, a few transit stops, grocery stores, and schools. The same percentage can be compared to another neighbourhood or another plan for the same neighbourhood. This way development plans can be ranked as to their aptitude for walking or cycling in terms of basic potentials. If a finer detailed comparison is desirable, then other aggregates such as average fuzzy closeness values of the POI in question can be compared from one case to another. The noteworthy advantage of fuzzy indicators is that every map/plan can be compared to another objectively. Having 'a walking travel time less than 5 minutes' to a (few) POI is a quality that is universally understandable both for citizens and planners. The function needed for generating such reports is simply the arithmetic average of the metrics attributed to plots. As an example, in the case of POI listed above, we can determine that:

- 85.1% of plots have a **walking access** within 10 minutes to ALL POI
- 76.4% of plots have a **walking access** within 3 minutes to ANY POI
- 96.7% of plots have a **cycling access** within 10 minutes to ALL POI
- 88.5% of plots have a **cycling access** within 3 minutes to ANY POI

Using fuzzy closeness measures, we can elaborate the above synthesis as:

- 13.4% of plots have a **good walking access** within 10 minutes to ALL POI
- 25.2% of plots have a **good walking access** within 3 minutes to ANY POI
- 22.8% of plots have a **good cycling access** within 10 minutes to ALL POI
- 32.9% of plots have a **good cycling access** within 3 minutes to ANY POI

The above results correspond to the exemplary POI introduced in Table 6 and Figure 167. The figures corresponding to these syntheses are shown in Figure 170.

We have chosen a few known locations in Tarlabasi, Istanbul as exemplary POI introduced in Table 6. There are 449 street segments in our simplified and generalized network (Tolerance=5 meters, Resolution=40 meters, Tau=30 Seconds).

POI_NAME	POI_#	HOW FAR (TO ANY) MINUTES	HOW FAR (TO ALL) MINUTES
Süryani Kadim Meryem Ana Kilisesi(The Virgin Mary Assyrian Church)	0	3	10
Çelebi Süpermarket	1	3	10
Yusufpasa Suites (Hotel Apartment)	2	3	10
Rum Ortodoks Aya Konstantin Kilisesi (Rum Orthodox Church)	3	3	10

TABLE 6 A list of real-world points of interests for the Tarlabasi dataset

FIGURE 167 shows the location of the POI on the sample neighborhood map (Tarlabasi, Istanbul) and their exclusive catchment zones according to the table above

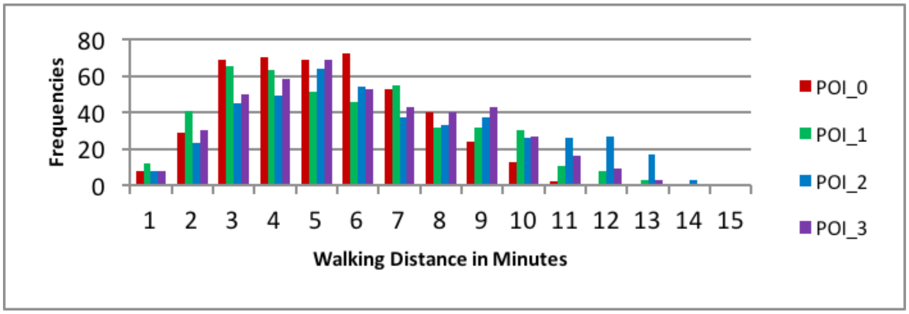


FIGURE 168 A histogram of walking distances to POI, colored as to zone colors above

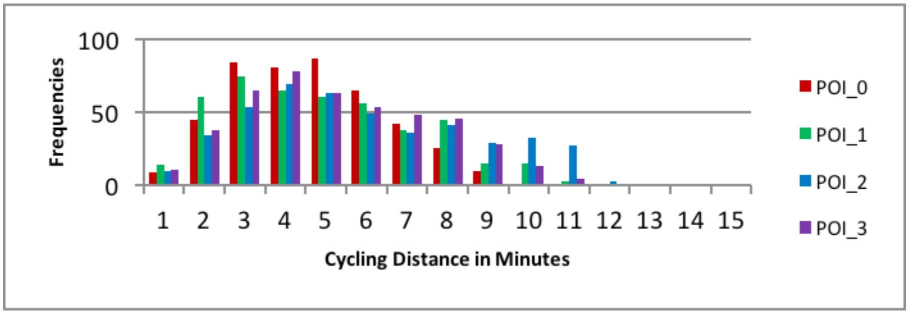


FIGURE 169 A histogram of cycling distances to POI, colored as to zone colors above

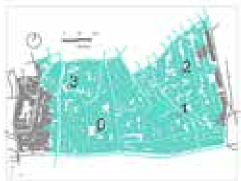
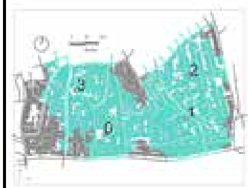
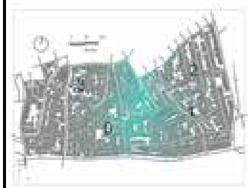
	Crisp		Fuzzy	
	To ALL (10 Min)	To ANY (3 Min)	To ALL (10 Min)	To ANY (3 Min)
Walking	85.1% 	76.4% 	13.4% 	25.2% 
Cycling	96.7% 	88.5% 	22.8% 	32.9% 

FIGURE 170 pictures of accessibility assessment results

§ 6.9 Qualitative Evaluation

Here we re-evaluate the usefulness of the toolkit regarding its purpose and its target group. Before judging it as a product in itself, however, it is necessary to acknowledge its role as a means of testing and improving the methodology introduced in the previous chapter.

We have revised the structure of the methodology as a computational suite of methods a few times. The toolkit CONFIGURBANIST has been publicly available since 2013, tested in four international workshops. We have received continuous feedback from an international user group (from the workshop audiences and group members on the tool's website) and improved the functionality and interface of the tools accordingly.

From the experience of test and development of the methodology, we can conclude the following:

- It was necessary to implement and test the mathematical methods computationally. Implementing many well-known methods from scratch might seem as a waste of time; however, it proved to be a good practice as it helped in developing novel methods. This is because design requires iterative formulations of both the problem and its solution.
- Research and Development proceed best when jointly conducted.
- In order to handle spatial networks it is convenient to work in an environment where doing geometric edits is easy. This is because network editing is an inherently interactive action. If our target user group is "urban designers", then choosing a design platform as the base environment is natural.
- In their current form, most of the methods produce results, which are potentially interesting for public; therefore, a cross-platform web-based implementation would be beneficial.

The last point regards our direction for the future developments of the methodology. We see potential application of the measurements done with the methods in our toolkit to be applied in such diverse areas as real estate, business intelligence, urban planning (and urban geography) and urban design (and urban morphology). This outlook demands for a platform that would be usable by a much wider user group.

§ 6.10 Future Work

The toolkit in its current form is not exactly user-friendly for its ultimately intended user-group, i.e. those decision-makers whose decisions can affect walking and cycling potentials in cities. Improving the user experience requires a paradigm shift from desktop applications to web-applications. This would ensure usability for a much wider user group and potentially simplify the workflows. There is however no either-or choice to be made. The library of methods can be improved in terms of its structure to be used in CAD and GIS applications and at the same time a parallel web-based platform can be developed for public users who might have some interest in assessing walking and cycling accessibility of their properties, businesses, neighbourhoods and alike.

A suitable infrastructure and technology platform for developing a web-based tool, should ideally allow for flow-based programming. This programming paradigm is currently flourishing with the advent of some new web-based programming technologies, among which we are searching for the most effective future platform to host CONFIGURBANIST.

7 Conclusions

This dissertation reported a research, development, and innovation process. We have not conducted empirical research but rather theoretical research as in developing mathematical and computational models and implementing them as well as (re)inventing tools for applied Spatial Network Analysis in Architectural Design and Urban Design.

This chapter:

- refocuses on the initial research questions;
- summarizes our research findings and achievements;
- mentions the limitations of the proposed methods;
- discusses the nature of our scientific contributions; and
- reflects on the future research areas

§ 7.1 Summary of Results

Here we summarize the results of this project from each chapter of this book.

§ 7.1.1 Results of Chapter 1

In this chapter, we introduced the basis of our methodology as a Design Science Research Methodology (after (March, Salvatore T., and Gerald F. Smith, 1995) and (Peffers, K, Tuunanen, T, Rothenburger, M A, Chatterjee, S, 2007)) and concluded that our Research, Development, and Innovation process should proceed through the following steps:

- Theoretical Reflection (on how methodically spatial performance can be addressed in architecture and urban design)
- Problem Formulation and Concept Development (for the configurational design and analysis methodologies)
- Mathematical Modelling
- Algorithm Design
- Software Development
- Verification
- Crowd-Sourced Test and Validation

§ 7.1.2 Results of Chapter 2

In this chapter, we reflected on the theoretical underpinnings of our methodological approach, discussed the particularities of design problems as compared to engineering problems, discussed the necessity of advanced spatial analytics in analysing spatial performance, and concluded that design is an intellectual activity that can be supported by computational methods in analysis and synthesis of configurations. We argued that automated evaluation might be problematic especially in dealing with social aspects of design. We reasoned that computational design methods could be useful in problem solving but the main intellectual aspects of design activities pertain to problem formulation and interpretational activities, e.g. in performance evaluation; that is, designers can be supported for making 'their own decisions' based on contextual information. The chapter concluded with an elaborated account of our Research, Development, and Innovation (RDI) methodology, as briefed in chapter 1.

§ 7.1.3 Results of Chapter 3

This chapter provides the mathematical and algorithmic basis of a new way of designing buildings that is based on spatial configuration. The methods developed provide for obtaining building layouts that fulfil programmatic constraints and requirements encoded in bubble diagrams. The basic idea of the methodology is to obtain concrete geometrical layout patterns from abstract configuration graphs, while providing real-time spatial network analysis (e.g. Space Syntax) to help the designer see the likely spatial performance of what they propose, so that they can explicitly focus on realizing the desired spatial qualities. Main innovations introduced in this chapter:

- An algorithm to interpret configurational sketches as configuration graphs
- Adaptation of an algorithm for automated drawing of bubble diagrams
- An algorithm for interactive drawing of justified graphs
- Adapting Space Syntax indicators to architectural configuration analysis
- Adapting Tutte graph drawing algorithm to form a basis for topological embedding of configuration graphs for plan layout
- An algorithm for cataloguing all topological plan layout possibilities for 2D plan layouts
- An algorithm for drawing alpha shapes for spatial configuration
- A computational geometry construct called 2D Isovist Bubble
- A computational geometry construct called 3D Isovist Bubble

§ 7.1.4 Results of Chapter 4

This chapter introduced the computational implementation of the methodology introduced in chapter 3, that is the toolkit SYNTACTIC for architectural configuration. Providing the toolkit as freeware has provided the opportunity for crowd-sourced test and validation of the methodology.

§ 7.1.5 Results of Chapter 5

This chapter provides the mathematical and algorithmic basis of a new way of analysing built environments regarding the effect of their spatial configuration on walking and cycling accessibility and mobility potentials. The methods developed provide for measuring polycentric accessibility considering an all-inclusive approach to modelling distance based on the notion of Easiest Path. The basic idea of the methodology is to provide objective insight into the ways in which spatial configuration could potentially facilitate or hinder walking and cycling potentials. Main innovations introduced in this chapter:

- An algorithm to interpret road centreline networks as configuration graphs which encode physical difficulty of walking/cycling and cognitive difficulty of navigation
- Easiest Path algorithm for finding paths that are as ‘straightforward, short, and flat’ as possible (cognitively and physically easiest paths for walking and cycling)
- A family of Fuzzy Logics aggregation methods for polycentric accessibility analysis
- Algorithms for finding zones of preferred access (generalized Voronoi and Alpha Shapes on network spaces)
- Adaptation of a family of geodesic network centrality models
- Adaptation of a family of spectral network centrality models
- Developing a family of probabilistic models for walking and cycling mobility based on Markov Chains

§ 7.1.6 Results of Chapter 6

This chapter introduced the computational implementation of the methodology introduced in chapter 5, which is the toolkit CONFIGURBANIST for urban configuration analysis. This chapter explains a number of subtleties and challenges in implementation of the models and algorithms introduced in the previous chapter, and discusses the foundation of configraphics.dll as the shared library of methods for spatial configuration analysis.

§ 7.2 Response to Research Questions

Here we look back at the initial research questions and see to what extent they are answered and what would change if addressing them again. Our initial research questions were the following:

How can we model spatial performance in architecture and urban design?

In response to the main question, we have developed a Spatial Configuration Graphs library (configraphics.dll, WIP), which encompasses methods for analysis of spatial configurations in architecture and urban design. This library has a mathematical/computational basis that treats architectural and urban spatial configurations in a similar way.

– How can we obtain an architectural layout from a spatial configuration graph, while controlling its performance? (Chapters 2, 3, 4)

We answered this question by developing a methodology for systematic synthesis of spatial layouts that is equipped with real-time spatial configuration analysis methods of Space Syntax. The methodology is based on gradual concretization of bubble diagrams drawn by the designer, while at the same time providing feedback on the likely spatial performance of the architectural configuration graph. The process put forward by this methodology currently ends in a catalogue of topological layout possibilities for 2D plans. In addition, we have provided smart computational geometry agents called Isovist Bubbles as a basis for Agent-Based Models for spatial layout in 2D and 3D.

The models and methods developed in this track of the project are from Topology (constructing configuration graphs from sketches), Graph Theory (the basis of all configurational analyses and the backbone of all methods), Computational Graph Drawing (Force-Directed Graph Drawing, Tutte Graph Drawing algorithm and Spectral Graph Drawing), Computational Topology (Alpha Shapes), and Computational Geometry (Isovist Bubbles 2D & 3D).

– How can we model the effect of spatial configuration on accessibility (e.g. by walking and cycling) and mobility potentials? (Chapters 2, 5, 6)

We answered this question by developing an urban configuration analysis methodology, which is thematically focused on modelling the effect of urban configuration on walking and cycling accessibility. This methodology is centred on a novel notion of distance that is based on our optimal path algorithm called Easiest Path. The Easiest Path algorithm finds the paths that are both physically and cognitively easy to walk or cycle; for which it considers the topography, topology, and the geometry of environment at the same time. The distances measured as to Easiest Paths form the basis of calculation of measures of accessibility of locations encoded as Points of Interests (POI). We model accessibility as potentially perceived by pedestrians and cyclists. In addition, we provide a novel family of methods for stochastic modelling of walking and cycling flows.

The models and methods developed in this track of the project are in the areas of Topology (constructing graph models from road centrelines), Graph Theory (the back bone of spatial configuration analytics, used in graph traversal and path-finding, e.g. Easiest Path algorithm), Fuzzy Logics (the polycentric accessibility framework), Spectral Graph Theory (spectral centrality measures), and Markov Chains (four Random-Walk models).

– How can we integrate architectural and urban spatial analyses and estimate the spatial performance of design proposals? (Chapters 6, 7)

We have sought the answer to this question by providing a basic integration of the computational methods developed separately for tackling architectural and urban configuration. There are similarities and differences in the geometrical, topological, graphical, and spectral properties of architectural and urban configurations.
Our definition of the spatial configuration graph is mathematically the same for architectural and urban configurations. We envisage that by integrating the two computational libraries of SYNTACTIC and CONFIGURBANIST in configraphics.
dll we can bring the two methodologies closer to each other for analysing spatial configurations. This area of work still requires further research and development.

§ 7.3 Discussion

There is a relative scarcity of clearly (mathematically) defined scientific research methods for assessing spatial performance of buildings (programmatic efficiency, social aptitude and provision for desired communal behaviours) and built environments (walking and cycling accessibility. We have focused exactly on the research methods themselves, as to which we have also questioned many common research questions and reformulated them. Compared to the common approaches to the issues tackled, we have done less of empirical testing and more of mathematical and computational development and tried to make the mathematics behind our methodology as clear as possible.

We believe that there is a need for developing exact sciences in the field of built environment and in doing so mathematics as the language of exact sciences would have the key role in communication of knowledge. However, the role of mathematical models should not be considered as crystal balls to predict how exactly the built environment configuration affects spatial behaviour. We need to clarify what can be understood from analysing networks. Not everything is representable on networks. Spatial Networks definitely affect mobility potentials and accessibility in general. Mobility and accessibility in turn affect the potentials for social interactions, e.g. certain network configurational situations might foster social integration or exclusion. However, there is always a multitude of other factors affecting spatial behaviour, which might not be necessarily representable on networks.

Knowledge of how exactly spatial configuration affects social configurations tends to be situational; therefore having re-configurable research tools would be advantageous. Although there might be many situational factors involved in mobility potentials and the effect of spatial configuration on social structure, we can focus on the objectively universal aspects of Spatial Network Analysis. This is exactly the approach we have adopted; i.e. to measure what is undisputedly measureable and provide a framework for measuring spatial distance and its effects on accessibility.

In architecture, the spatial layout process can be seen as a process of going from very abstract functional requirements to concrete geometric drawings of spaces. In the abstract ambience of early design stages, the notion of spatial distance is of graphical (pertained to Graph Theory) nature; thus a graph centrality approach such as that of Space Syntax could serve as an analytic module necessary for reflecting on spatial performance in design process. Evaluation of spatial performance, however, cannot be easily automated, as it tends to be context-based and situational.

The main challenge in approaching spatial layout explicitly through configuration is on one hand the sophisticated computational topology required for enumerating the embedding possibilities, and on the other hand the complicacy of the computational geometry constructs and algorithms required for realizing a potentially 3D embedding into a concrete geometric form. This area of research has received much more attention in computer science in such areas as graph drawing algorithms, floor planning of Integrated Circuits and Geomatics (as in production of rectangular maps as simplified versions of geographical maps).

The process that we have developed so far for configurational layout can be seen as a theoretical investigation that can strengthen the foundations of a science of design, i.e. an approach that was pioneered by scholars such as March and Steadman (March, L, Steadman, P, 1974).

Our main contribution in the area of configurational analysis can be seen in chapter 5, in laying new foundations for measurement of distances, the fuzzy concept of accessibility, and the demystification of spectral network analytics and development of a new family of biased random walk models for stochastic modelling of mobility patterns.

§ 7.4 Future Work

We have mentioned the limitations of our methodologies and the areas of future work in detail at the end of each corresponding chapter and. Here we highlight the most essential areas that need deeper investigation in future.

§ 7.4.1 Configurative Architectural Layout (Chapters 3, 4)

The approach we have introduced in chapter 3 is mainly a foundation for a completely configurational design process. There are still several challenging topics pertained to configurative designs that require further theoretical research on mathematical foundations and algorithmic methods, namely:

- How can we enumerate the number of ways a graph can be topologically embedded in 3D? (Computational Topology)
- How can we obtain a 2D plan layout using 2D Isovist Bubbles and a 2D topological map?
- How can we obtain a 3D plan layout using 3D Isovist Bubbles and a 3D topological map?

The above questions address the synthesis phase of the configurative design process. The analytic aspect of the process can be further strengthened by adapting the methods introduced in chapter 5 to the analysis of networks in architectural scale. Any set of analytic models could then be validated by studying the actual spatial performance of buildings.

§ 7.4.2 Spatial Configuration Analysis (Chapters 5, 6)

The analytic methodology introduced in chapter 5 can be potentially extended to include more contextual parameters (e.g. densities and land-uses) for predicting network flows. After that, a new generation of a all-inclusive models can be developed for predicting walking and cycling flows. These models can then be validated and calibrated based on the actual movement patterns of people. The major area of work can therefore be identified as incorporation of land-use and population density in the models. Developing the methodology in the context of cases where neighbourhoods are to be developed in vacant lands can be potentially insightful for combining the two methodologies SYNTACTIC and CONFIGURBANIST.

Endnotes

Chapter 1

1 Architecture is interdisciplinary in that it involves dealing with structural stability, climatic comfort and alike but what is essentially 'architectural' is shaping spatial structures in order to serve some kind of a function. Herman Hertzberger beautifully termed architects as "spatialists" in his inspiring speech at INDESEM 2015 at TU Delft. Nobel laureate Herbert Alexander Simon, in his famous book the Sciences of the Artificial (Simon, 1999, pp. 132-135) wrote: *"Since much of design, particularly architectural and engineering design is concerned with objects or arrangements in real Euclidean two-dimensional or three-dimensional space, the representation of space and of things in space will necessarily be a central topic in a science of design. From our previous discussion of visual perception, it should be clear that "space" inside the head of the designer or the memory of a computer may have very different properties from a picture on paper or a three-dimensional model."* We are requoting this after Philip Steadman (Steadman, 1983)for a similar reason.

2 For measuring socio-spatial performance, we rely on theories that can relate social behavioural patterns in built space to spatial configuration; i.e. considering how a certain configuration facilitates certain encounters and hinders others via influencing accessibilities.

3 Urban transportation with private cars forms a significant part of this consumption trend and has increased in the recent 50 years continuously. Exact figures and statistics are not mentioned because sources are many and the issue falls out of scope of this research.

4 The basic definitions of Space Syntax of visibility in fact show aspects in the theory that require reworking on cityscapes with considerable hilly terrains.

5 see for instance the problem with Diamond Value computation explained by Hoon Tae Park

6 The kinds of SDSS discussed in this paper are not exactly applied in architectural or urban design. Nevertheless, the same issues mentioned hold for the so-called design systems developed for instance for architectural layout, such as those introduced (Lobos, D., Donath, D., 2010).

7 The actual research process is in fact not as neat as illustrated here. The process has been a research and development cycle as compared to mere research. There have been many failures be it technical, methodical or those pertained to application area of research findings. There have been many detours, serendipities, and iterations as well.

8 This DLL was initially conceived as a kernel for a Planning Support System for promoting walking and cycling and informing decisions on their walking/cycling accessibility impacts. This PSS was proposed in the context of a Horizon2020 proposal initiated by the author together with Dr. Frank van der Hoeven for the European Commission research grant call: Mobility For Growth MG 5.3 2014. Our proposal was to develop and disseminate an evidence-based computational approach to spatial planning for improving walking and cycling to reduce urban road congestion: http://ec.europa.eu/research/participants/portal/desktop/en/opportunities/h2020/topics/2638-mg-5.3-2014.html
The proposal was evaluated by peers in two rounds, initially as excellent in the first round (in terms of Scientific and Technological Excellence, and Expected Impact). The second stage proposal was evaluated as good (Scientific and Technological Excellence: Good, Expected Impact: Good, Work Plan: Good). Involved parties included the following:

PARTICIPANT NO	PARTICIPANT ORGANISATION NAME	ABBREVIATION	COUNTRY
1 Lead	*Technische Universiteit Delft (TU Delft)*	TUD	The Netherlands
2	*University College London (UCL), The Bartlett*	UCL	United Kingdom
3	*PBL Netherlands Environmental Assessment Agency*	PBL	The Netherlands
4	*Space Syntax Ltd*	SSX	United Kingdom
5	*Transport Insights Ltd*	TIS	Ireland
6	*Open Sky Data*	OSD	Ireland & Poland
7	*Câmara Municipal de Lisboa*	LIS	Portugal
8	*Municipality of Ljubljana*	LJU	Slovenia
9	*Samenwerkingsverband Regio Eindhoven (SRE)*	SRE	The Netherlands
10	*European Council of Spatial Planners ECTP-CEU*	ECTP	Belgium, EU

The parties are mentioned as a courtesy to their contribution to the development of some ideas in the project’s proposal, which have influenced our work. Their mention in this work does not imply their involvement in or endorsement of this dissertation.

9 We use the term model in so many different contexts that it has almost no specific meaning anymore, unless we specify using prefixes such as geometric model, statistical model, stochastic model and alike. Here in this dissertation, we use the term model generally in the sense of a mathematical or computational model with inputs and outputs. Specifically, by modelling a system we mean providing a mathematical formulation or computational system that can mimic a certain behavioural aspect of the system in question or indicate something about its performance. In this sense, we should distinguish two different kinds of models: those that are a 'model of some system', and those that are a 'model for a process'. We offer both types of models in this work; their meaning will be clear in their context.

10 Design Research as such addresses issues in a multi-disciplinary and trans-disciplinary manner. It is about the nature of design activities and the shared characteristics of design professions in a variety of professions ranging from product design to urban design.

11 The term as used by Lawson can be somewhat confusing and problematic. In the conventional scientific sense, synthesis is referred to as the process of putting together analyses in order to reach at a conclusion. However, design researchers seem to have adopted a different meaning and created a jargon-like sense for the term referring to 'generation of design alternatives'. It would have been better to avoid this terminology; however, it is already widely used by design researchers.

12 Lawson has also emphasized the fact that design thinking is more solution-oriented than problem-oriented as compared to the scientific thinking. He also states that in most design cases synthesis comes first (Lawson, 2004).

13 By taking account of such phenomena as measured in terms of 'states' of spatial systems (e.g. the number of people passing through or present at a space) it can be readily seen that we are not dealing with deterministic models but with rather stochastic phenomena and models.

14 The exact physical definition can be found in standard systems text books such as (Ogata, 1987), the definition simply defines a system whose current 'state' does only depend on its past and current inputs but not on future inputs and that it does not have any information about its future. Human decision makers are definitely not so, for they have clear ideas and wills about future and they hold certain actions for their potential consequences and that they plan for future. In other words, we humans have some reflexive actions but we usually have reasons for many of our actions, not causes; we just do not do things as reactions to other things.

15 This is in fact a very delicate subject in need of deeper clarifications. Self-Organization and Self-Regulatory processes caused by feedback mechanisms have in fact governed the morphogenesis or many rural/ vernacular settlements; in view of sustainability we could say such settlements would have been mostly sustainable and in harmony with needs of inhabitants and surrounding nature. Such settlements can be distinguished by the fact that their formation has been gradual and unplanned rather than being centrally planned or designed. However, there are also many evidences of large-scale urban interventions or even construction of new cities by the order of kings, rulers, and politicians, many of which have failed throughout history. Design and planning can well be a political act, which does not necessarily comply with the view of city as a self-organizing system possessing self-regulatory mechanism. One interesting question would be what could we learn from the self-organization of vernacular settlements to apply to the planning and design process of modern settlements?

16 Proprietary CAD software applications such as:
Autodesk's AutoCAD http://www.autodesk.com/products/autocad/overview
McNeel's Rhinoceros3D https://www.rhino3d.com/ ; and
Bentley's MicroStation: http://www.bentley.com/en-US/Products/MicroStation/

17 Environments such as Autodesk AutoCAD could run macros in LISP programming language and alike. With macros (routines) one could potentially automate certain cumbersome tasks involving repetition of a series of instructions; but processing flows of data and setting up more intricate processes is not straightforward in this approach.

18 Freeware Add-on applications such as:
Bentley's Generative Components: http://www.bentley.com/en-US/Promo/Generative%20Components/default.htm
McNeel's Grasshopper3D: http://www.grasshopper3d.com/
Autodesk's Dynamo-Vasari http://autodeskvasari.com/dynamo

19 Flow-Based Programming is a relatively new paradigm in programming in which flows of data are being processed by modules of code or functions with multiple inputs and multiple outputs. There are presently many general-purpose FBP environments such as NoFlo for Javascript, and of course environments dedicated to geometric computing and computer aided geometric design such as Grasshopper 3D and Dynamo. J Paul Morison is a pioneer and an outspoken advocate of this new paradigm in programming.

20 To illustrate the point, let us suppose a case, where we are to draw a circle of surface are equal to 100 m2. One way to do this would be to first draw a circle of arbitrary size, and then measure its surface area and then play with the radius so as to find the radius that gives the right answer. This is a feedback strategy. A much more reasonable strategy, in this case, would be to analyse the problem and observe that the area of a circle is determined by its radius with the relation $A = \pi r^2$, so if we are to draw a circle

of a certain surface area, then we can find the right radius $r = \sqrt{A/\pi}$ to draw the right circle immediately. This is a feedforward strategy. We advocate the feedforward strategy wherever feasible for it is explicitly analytic and efficient while providing full control on the course of design.

21 Widely-accepted definitions are rare. Most definitions are inclined towards defining spatial analysis as spatial-statistics, or leaning on reduced definitions of geometric/topological operations involving spatial data queries. There are also some definitions like that of us that sound like tautologies of the terms.

22 Such as those usually encoded in Dimensionally Extended nine-Intersection Model (DE-9IM) implemented in ISO and OGC (Simple feature access) standards, e.g.: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=40115

23 More information and examples on http://wiki.gis.com/wiki/index.php/Spatial_Decision_Support_System

24 To this category, perhaps we should add the Land-use Transportation Interaction Models (LUTI). An introduction to these models can be found in (Torrens, 2000), & (Dios Ortuzar, J., & Willumsen, L. G., 2011). The subject, however, falls out of scope of this research

Chapter 3

25 As most researchers use simple or small graphs as examples to explain the idea of using centrality measures for analysing architectural qualities some people think these measures only confirm what architects know intuitively. Indeed, if a theoretical measure produces a ranking, which seems to be intuitive, then it is useful in studying larger cases where the patterns cannot be immediately seen intuitively.

26 It must be noted that many people confuse the notion of space represented within an n-dimensional 'feature' (such as points, curves, or surfaces) with the space in which they are 'embedded'. This confusion is caused by the fact that features (objects such as points, lines, or surfaces) are often embedded, drawn, or visualized in 3D Euclidean space. However, the features themselves represent a space within which every point can be addressed using n numbers, so to speak. This means the feature under study (only manifold features in our work) topologically resemble (i.e. they are locally homoeomorphic to) the Euclidean space of dimension n. For example, the surface of the globe can be considered a 2D space (only by imaging from above) in which every location can be mapped in terms of two dimensions such as longitude and latitude. In this case, the atlas constructed as such would be homoeomorphic to a rectangle and both surfaces locally resemble the Euclidean 2D space, i.e. they are 2-manifolds. This is a reason why -for many years- people conceived of Earth to be flat.

27 To be precise, what is meant by Topology is Topological Graph Theory.

28 Meshes contain 0D, 1D and 2D elements, respectively called vertices, edges, and faces. It is conventional to store mesh topology in terms of adjacency or incidence information between these elements considering the mesh as an embedding of an undirected graph:

	V	E	F
V	VV	VE	VF
E	EV	EE	EF
F	FV	FE	FF

There are also alternative data-structures for storing meshes such as Winged-Edge and Half-Edge. The whole subject is outside of the scope of this thesis but interested readers can find more about this subject in computer graphics text books.

29 The complete equation is $\chi(M) = |V|-|E| + |F| = 2(1-g) - \partial$ where g is the number of genera (pl. form of genus) is and ∂ is the number of boundaries composed of open edges. This chcarcteristic is equal to 1 for maps (meshes) representing 2-manifold surfaces.

30 That is a toolkit consisted of the methods and algorithms introduced in this chapter. The computational design toolkits (SYNTACTIC and CONFIGURBANIST) are currently available for the Grasshopper3d parametric design platform. They can be adapted to other platforms by means of a shared library of computational methods called configraphics.dll, that is under development.

31 The Diamond Value is to make Integration Values comparable irrespective of the size of the network. Its formulation has been revisited thoroughly by Hoon Tae Park in (Park, 2005).

32 We did not find a particularly interesting criterion for this selection to be generalized. It seems best to leave this to be defined in a design situation. Note that all alternatives would be the same in terms of their provision for the desired spatial connectivity, but they will be different in terms of the undefined adjacencies. Adjacencies would be more important in such areas as climatic or structural design perhaps.

33 We are now considering representing the actual border of the plan and its vertices instead of these vertices. However, such a border has either to be a convex border or be projected to its own convex hull in order let the Tutte barycentre algorithm work. In any case, the NEWS vertices can be considered as a very generic border of a plan, given unknown design situations.

34 As mentioned before, we have decided to focus on an alternative path starting from topological plan layout patterns, which involves using isovist bubbles.

35 Note that in this generic set up of the design process we do not have any information to judge these distinct possibilities. However, as the tools can be easily integrated with other computational design workflows, a designer can develop a project specific search in the catalogue of possibilities, if necessary.

36 This algorithm, not to be mistaken with Voronoi Diagram, does not exist as such in Grasshopper. This is our own implementation. All other results are also achieved using our own implantations.

37 It must be noted that the algorithm for finding all possible topological embedding patterns in 2D was already one of the most challenging and sophisticated parts of this work. The extension of the method to 3D embedding would be even more significantly challenging.

38 Instead of the nominal NEWS sides for a plan, any other convex polygon can be used in principle but that requires adaptation of the algorithms already implemented

Chapter 4

39 The library configraphix.dll (written in C#) is already used in the new version of CONFIGURBANIST. It mainly contains methods for computation of easiest paths at present. It will be released as soon as a stable versions is ready on the author's website: https://sites.google.com/site/pirouznourian/home

40 We could potentially combine most of the tools in a single box; however, we chose to let the curious user be able to explore different possibilities for combining the tools from a toolkit.

41 The main tools shown in the images of this chapter are from either SYNTATIC toolkit or other components made by the author, except for data wrappers, which are native tools of GH.

42 http://www.gbl.tuwien.ac.at/Archiv/digital.html?name=SpiderWeb

43 http://www.decodingspaces.de/content/decoding-spaces-components-grasshopper-rhino

44 http://www.archtech.gr/varoudis/http://www.archtech.gr/varoudis/?tag=depthmap?tag=depthmap

45 The definition of a good graph drawing is indeed very extensive but it suffices to say that a good graph drawing must at least have good geometric resolution and distinction between nodes.

46 A quote from a computer-graphics expert, whose reference we cannot recall or find unfortunately!

Chapter 5

47 It is known that simple path finding algorithms pose limitations even in modelling transport (Dios Ortuzar, J., & Willumsen, L. G., 2011, p. 381)

48 Two generic types of raw datasets can be used to model the spatial network, namely vector and raster datasets. In vector datasets there are 0D features (points), and 1D features (lines or polylines) possibly also 2D features (polygons).

49 For a more in-depth comparison of models we refer the reader to (Batty, 2004)

50 The graph model of bridges of Konigsberg by Leonhard Euler (1707-1783) had bridges represented as links and lands as nodes

51 They consider the so-called "natural movement" (Hillier, B., Penn, A., Hanson, J., Grajewski, T. & Xu, J., 1993)

52 This discussion tends to be very problematic and extensive. We refer the reader to the paper of Ratti (Ratti, 2004) that triggered a long discussion on the problems of reducing built environment to Space Syntax models. We could mention points for or against both stances in this discussion but that would have taken so much space from the already long chapter. Anyhow, it suffices to mention that we should critically think of what we are representing in a model and manage our expectations on what can be understood from a model according to its scope and limitations.

53 We form a Point-to-Line incidence matrix which when transposed makes the Line-to-Point incidence matrix. When multiply a P2L adjacency matrix by itself transposed (L2P) we obtain a Point-to-Point adjacency matrix, and similarly when multiplying a L2P incidence matrix by itself transposed we obtain a Line-to-Line adjacency matrix.

54 Previously, they have been categorised as primal and dual representations. We consider this distinction merely as a meta-label to categorise two large lines of works. However, the terms do not bear the exact meaning of Poincare duality in topology. In fact, what we are interested in is the distinction between spatial network models that model the connection between spaces as links from those, which model spaces as links between spatial junctions.

55 The graph model of bridges of Konigsberg by Leonhard Euler (1707-1783) had bridges represented as links and lands as nodes.

56 The spatial network representation of Space Syntax graph that is based on axial lines is in fact a particular class of visibility graphs. Further discussion on this topic is not in the scope of this paper.

57 We have later defined admittance weights in the context of weighted adjacency matrices used for constructing probabilistic models. Impedance weights are convenient for geodesic algorithms but would be meaningless or counter-intuitive in the context of graph-drawing, spectral graph theory or Markov chain models.

58 Physical quantities can have different units but they are all made up of seven fundamental types of quantities as Electric Current (A), Length (L), Time (T), Mass (M), Absolute Temperature (K), Luminous Intensity (C), and Amount of Substance (Mol). Other quantities can be shown to be having combinations of these dimensions. For instance, the dimension of force is MLT-2. If two quantities have different dimensions, then addition and subtraction as well as making arithmetic averages between them would be physically meaningless! 2 Euros+2 Meters does not make any physical sense. Although we can mathematically put 2 and 2 together and get 4 but that does not make any physical sense in this case.

59 The details of these models are discussed in a forthcoming book chapter in Research in Urbanism Series, to be published by TU Delft, faculty of Architecture and the Built Environment.

60 It might seem easy to add angles to meters of distance but that would be physically wrong, just like adding peers and apples. When two quantities are added, they need to have the same physical dimensions and units. This issue is referred to as unit commensurability in physics.

61 Finding the link index (k) of for the link $\mathrm{L}_{\mathrm{i,j}}$ we can get the cost of each link from the pre-calculated impedance set.

62 A path π is a sequence of nodes $(n_0, n_1, n_2, \ \ldots, n_n) \in N \times N \times \ldots \times N$ [where N is the set of nodes in the graph and $\times$ denotes Cartesian product], such that $n_i \sim n_{i+1}$ for $i \in [0, n]$

63 terminology and definition from (Borgatti, Stephen P., Everett, Martin G., 2006)

64 The entirety of subject is well addressed in the textbook by Chung (Chung, 1997).

65 A common notation, which in our view is more obscuring than helpful, is to show a weighted graph as an ordered pair $\Gamma(\mathrm{N}, \mathrm{L}, \mathrm{w})$, in which weights are indexed as to the links

66 Chapman-Kolmogorov equations in this case will be applied to transition probability matrices $\boldsymbol{P}^{n+m} = \boldsymbol{P}^n \boldsymbol{P}^m$, which actually mean the following: Probability of being at state j at time $n+m$ is the (i, j) entry of the transition probability matrix raised to the power of $n+m$ because the discrete version of Chapman-Kolmogorov equations say:

$$Pr(\boldsymbol{X}_{n+m} = \boldsymbol{j} | \boldsymbol{X}_0 = \boldsymbol{i}) = \sum_{k=1}^{n} Pr(\boldsymbol{X}_n = \boldsymbol{k} | \boldsymbol{X}_0 = \boldsymbol{i}) \, Pr(\boldsymbol{X}_{n+m} = \boldsymbol{j} | \boldsymbol{X}_n = \boldsymbol{k})$$

That is to say:

$$p_{(i,j)}^{n+m} = \sum_{k=1}^{n} p_{(i,k)}^{n} p_{(k,j)}^{m}$$

Which in matrix form looks like the following:

$$\boldsymbol{P}^{n+m} = \boldsymbol{P}^n \boldsymbol{P}^m$$

It follows from this equation that in order to fine the transition probability matrix at time n it suffices to raise the transition probability matrix of time 0 to the power n; of course if the process is time homogenous, i.e. the probabilities are the same over time, but not necessarily the same over space.

67 If there exists a power of the transition probability matrix at which the transition probability of i to j is positive; formally, $\exists n \geq 0 | p_{(i,j)}^{n} > 0$, i.e. the state j is eventually accessible from state i we say the Markov Chain is irreducible if every state is accessible from every other state. For more information and a formal treatment see for instance the lecture notes of Olivier LÉVÊQUE: http://ipg.epfl.ch/~leveque/LectureNotes/random_walks.pdf

68 A state i is aperiodic if the greatest common denominator of transition times at which the probability of transiting to itself is greater than zero is 1. Formally, the period of a state is defined as $d(i) = gcd\{n: p_{(i,i)}^{(n)} > 0\}$. What can happen for a periodic state is that it has a positive probability of returning to itself at even iterations and a zero such probability at odd iterations. For more information and a formal treatment see for instance the lecture notes of Olivier LÉVÊQUE: http://ipg.epfl.ch/~leveque/LectureNotes/random_walks.pdf

A more intuitive description of a period is the greatest common divisor of the number of times that the system has to go under transition such that a random walker would return to a state. In other words, we say a state i has the period k if an eventual return to state i must happen in 'multiples' of k, hence the definition:

$$k = \gcd\{n > 0: \Pr(x^{(n)} = i | x^{(0)} = i) > 0\}$$

If we expect that a return to state i can occur in 4th, 6th, 8th... iterations, then we say the period of state i is 2; although we cannot expect the random walker to return to state i in just two steps.

69 A state i is recurrent if the eventual probability of returning to itself for some iteration time larger than zero equals 1, i.e. it for sure returns to itself following a random walk, the only question is when it does so, not whether it does so or not. We call a state positive recurrent if the mean recurrence time a.k.a. the expected return time for that state is finite. Formally, if we define T_i as the first return time to state i:

$$T_i = \inf\{n \geq 1: X^{(n)} = i | X^{(0)} = i\}$$

The probability that we first return to the same state after n time is then denoted as $f_{ii}^{(n)}$:

$$f_{ii}^{(n)} = \Pr(T_i = n)$$

If the probability that the first return time is less than infinity (meaning that it would not take almost forever to return to the same state) is less than one, then the state is called transient, meaning once one passes it there is practically no chance to return to the same state. This is shown as below:

$$\Pr(T_i < \infty) = \sum_{n=1}^{\infty} f_{ii}^{(n)} < 1$$

This means that recurrent states (non-transient states) are guaranteed to have a finite return time; but we also need to ensure that the 'expected' number of iterations is finite in order to ensure ergodicity. The expected number of iterations (alias hitting time) for the eventual return is called Mean Recurrence Time denoted as M_i, and formulated as below:

$$M_i = \sum_{n=1}^{\infty} n.\Pr(T_i = n) = \sum_{n=1}^{\infty} n.f_{ii}^{(n)}$$

A state is called positive recurrent if M_i is finite.

For more information and a formal treatment see for instance the lecture notes of Olivier LÉVÊQUE: http://ipg.epfl.ch/~leveque/LectureNotes/random_walks.pdf

70 Do not mistake this directed graph with the graph constructed out of the doubly directed edges as explained under the title Dual Doubly-Directed

Chapter 6

71 The fact that the environment seems playful and friendly does not mean that writing a plugin software application for it is also a playful job!

72 For performance issues we are gradually replacing the matrix algorithms of MathNet by our own algorithms.

73 Map Generalization: Decreasing the level of detail on a map so that it remains uncluttered when its scale is reduced. Reference: http://support.esri.com/en/knowledgebase/GISDictionary

74 Spectral Modelling of Spatial Networks, the proceedings of 7th SimAUD symposium, UCL, London, pp. 103-110

75 Nevertheless, for our future work we have decided to improve efficiency by implementing the Johnson's Algorithm for all-pairs shortest paths or an optimized Dijkstra algorithm; therefore a change of graph data models back to adjacency lists (practically array or lists) would be necessary.

76 Iterative Methods such as Krylov Subspace methods .See for instance:
http://nmath.sourceforge.net/doc/numerics/MathNet.Numerics.LinearAlgebra.Sparse.Linear.html
and
http://nmath.sourceforge.net/doc/numerics/MathNet.Numerics.LinearAlgebra.Sparse.Linear.html

77 the other source codes might be published at https://github.com/Pirouz-Nourian

78 Methods in Object Oriented Programming that do not require an instance of an object to operate on are called static (in C#.NET) or shared (in VB.NET). These methods can be effectively used wherever needed without having defined an object.

Works Cited

Aichholzer, O., Aurenhammer, F., Alberts, D., & Gärtner, B., n.d. A novel type of skeleton for polygons. journal of Universal Computer Science, 1(12), pp. 752-761.
Allain, R., 2013. Dot Physics. [Online]
Available at: http://www.wired.com/2013/03/whats-the-steepest-gradient-for-a-road-bike/
[Accessed 2014].
Baglivo, J.A. & Graver, J. E., 1983. Incidence and Symmetry in design and architecture. Cambridge: Cambrdige University Press.
Banister, D., 2004. Sustainable transport and public policy. s.l.:EOLSS.
Batty, M, Rana, S, 2004. The automatic definition and generation of axial lines and. Environment and Planning B: Planning and Design, Volume 31, pp. 615-640.
Batty, M.& Torrens, P, 2001. Modeling Complexity : The Limits to Prediction. Caux, France, s.n.
Batty, M., 2004. A New Theory of Space Syntax. CASA Working Paper Series, March.
Batty, M., 2007. Planning support systems: progress, predictions, and speculations on the shape of things to come.. Cambridge, MA, s.n.
Benedikt, M. L., 1979. To take hold of space: isovists and isovist fields. Environment and Planning B, 6(1), p. 47 – 65.
Blanchard, Philippe, and Dimitri Volchenkov, 2008. Mathematical analysis of urban spatial networks. s.l.:Springer Science & Business Media.
Boehm, B., 1989. Software Risk Management. s.l., s.n., pp. 1-19.
Bonacich, P., 1972. Factoring and weighting approaches to status scores and clique identification.. Journal of Mathematical Sociology, 2(1), pp. 113-120.
Bonacich, P., 1987. "Power and centrality: A family of measures.. American journal of sociology, pp. 1170-1182.
Borgatti, Stephen P., Everett, Martin G., 2006. A Graph-theoretic perspective on centrality. Social Networks, 28(4), pp. 466-484.
Chung, F. R., 1997. Spectral graph theory Vol. 92. s.l.:American Mathematical Soc..
Cross, N., 1982. Designerly Ways of Knowing. Design Studies, 3(4), pp. 221-227.
Cross, N., 1999. Design Research as a Disciplined Conversation. Design Issues, 15(2), pp. 5-10.
Cross, N., 2007. Forty Years of Design Research. Design Research Quarterly, January, Volume 1, pp. 3-5.
Dalton, R. C., 2003. The Secret Is To Follow Your Nose Route Path Selection and Angularity. Environment and Behavior, 35(1), pp. 107-131.
Dios Ortuzar, J., & Willumsen, L. G., 2011. MODELLING TRANSPORT. Fourth ed. s.l.:John Wiley & Sons.
Dorst, K., 1997. Describing Design. Delft: Delft University of Technology.
Dorst, K., 2007. The Problem of the Design Problem. In: Expertise in Design - Design Thinking Research Symposium 6. Sydney: Creativity and Cognition Studios Press.
Dorst, K. & Cross, N., 2007. 'Co-evolution of Problem and Solution Spaces in Creative Design'. In: Computational Models of Creative Design. Sydney: Key Centre of Design Computing and Cognition.
Duckham, M., and Kulik, L., 2003. "Simplest" Paths: Automated Route Selection. s.l., Springer Berlin Heidelberg, pp. 169-185.
Duncan, W. R., 1996. A guide to the project management body of knowledge.. s.l.:s.n.
Eades, P., 1984. A Heuristic for Graph Drawing. Congressus Numerantium , 42(11), p. 149–160.
Eades, P., 2012. How to draw a graph, revisited. [Online]
Available at: http://www.csse.monash.edu.au/~gfarr/research/slides/Eades-HowToDrawGraphRevisit-edv11.pdf
[Accessed 1 8 2013].
Edelsbrunner, H., & Harer, J., 2008. Computational Topology. North Carolina: AMS American Mathematical Society.
European Commission, 2007a. Green Paper- Towards a new culture for urban mobility. Brussels: European Commission.
European Commission, 2011. WHITE PAPER Roadmap to a Single European Transport Area – Towards a competitive and resource efficient transport system, Brussels: European Commission Publication Office.
Fallman, D., 2003. Design-oriented Human—Computer Interaction. s.l., ACM, pp. 225-232.
Felkel, Petr, and Stepan Obdrzalek, n.d. Straight skeleton implementation. 1998, s.n.

Fidler, Dror & Hanna, Sean, 2015. Introducing random walk measures to space syntax. London, UCL.
Freeman, L., 1977. A set of measures of centrality based upon betweenness.. Sociometry, Volume 40, p. 35–41.
Gibson, J. J., 1977. The theory of affordances. In: The People, Place, and Space Reader. Hilldale, USA: s.n.
Gibson, J. J., 1979. The Ecological Approach to Visual Perception: Classic Edition. s.l.:Psychology Press.
Gil, J., 2014. Analyzing the Configuration of Multimodal Urban Networks. Geographical Analysis, 46(4), pp. 368-391.
Gould, P., 1967. On the Geographical Interpretation of Eigenvalues. Transactions of the Institute of British Geographers, Issue 42, pp. 53-86.
Habraken, N. J., 1988. Type as a social agreement. Seoul, s.n.
Hall, K. M., 1970. An r-dimensional quadratic placement algorithm.. Management science, 17(3), pp. 219-229.
Hanson, J., 1998. Decoding Homes and Houses. Cambridge: Cambridge University Press.
Hillier, B., & Ida, S., 2007. Network and Psychological Effects in Urban Movement. Melbourne, Springer.
Hillier, B., Burdett, R., Peponis, J., Penn, A., 1987. (1987), Creating Life: Or, Does Architecture Determine Anything?. Architecture et Comportement/Architecture and Behaviour, 3(3), p. 233 – 250.
Hillier, B., Hanson, J., 1984. The Social Logic of Space. 1997 ed. Cambridge: Cambridge University Press.
Hillier, B., Penn, A., Hanson, J., Grajewski, T. & Xu, J., 1993. Natural movement: or, configuration and attraction in urban pedestrian movement. Environment and Planning B: Planning and Desig, Issue 20, pp. 29-66..
Hillier, B., 2007. Space is the Machine. LONDON: Cambridge University Press.
Horvath, I., 2001. A CONTEMPORARY SURVEY OF SCIENTIFIC RESEARCH INTO. GLASGOW, s.n.
Horvath, I., 2004. A treatise on order in engineering design research.. Research in Engineering Design, 15(3), pp. 155-181.
Hurtado, F., Noy, M., 1996. Ears of triangulations and Catalan numbers. Discrete Mathematics, p. 319 324.
Hurtado, Ferran, and Marc Noy, 1999. Graph of triangulations of a convex polygon and tree of triangulations.. Computational Geometry, 13(3), pp. 179-188.
IEA (International Energy Agency), 2013. KeyWorld_Statistics_2015, s.l.: IEA (International Energy Agency).
Jiang B, Liu C, 2009. Street-based topological representations and analyses for predicting traffic flow in GIS. Geographical Information Science, 23(9), pp. 1119-1137.
Jiang, B, Claramount C, Klarquist, B, 2000. An Integration of Space Syntax into GIS for Modelling Urban Spaces. International Journal of Applied Earth Observation and Geoinformation 2.3 , pp. 161-171.
Jiang, B., & Claramunt C., 2004. Topological analysis of urban street networks. Environment and Planning B, 31(1), pp. 151-162.
Jiang, Bin, and Xintao Liu., 2010. Automatic Generation of the Axial Lines of Urban Environments to Capture What We Perceive. International Journal of Geographical Information Science 24.4, pp. 545-558.
Jiang, B., 2007. A topological pattern of urban street networks: Universality and peculiarity. Physica A: Statistical Mechanics and its Applications, 384(2), pp. 647-655.
Jiang, B., 2009. Ranking spaces for predicting human movement in an urban environment. International Journal of Geographical Information Science, 23(7), pp. 823-837.
Jiang, B., 2009. The Image of the City: From the Medial Axes to the Axial Lines.. Hanover, Germany, s.n.
Katz, L., 1953. A new status index derived from sociometric analysis. Psychometrika, Volume 18, pp. 39-43.
Koren, Y., 2003. On spectral graph drawing. In: Computing and Combinatorics. Berlin Heidelberg: Springer, pp. 496-508.
Kroes, Peter and Meijers, Anthonie, 2006. The dual nature of technical artefacts.. Studies in History and Philosophy of Science Part A, 37(1), pp. 1-4.
Laine, S., 2013. A Topological Approach to Voxelization. Computer Graphics Forum, 32(4), p. 77–86.
Lawson, B., 1980. How designers think. 4th, 2005 ed. Burlington: Architectural Press, Elsevier.
Lawson, B., 2004. Schemata, gambits and precedent: some factors in design expertise. Design Studies, 25(5), pp. 443-457.
Lee, J., 2001. 3D Data Model for Representing Topological Relations of Urban Features. San Diego, CA, s.n.
Lobos, D., Donath, D., 2010. The problem of space layout in architecture:.. arquiteturarevista, 6(2), pp. 136-161.
Maher, M. L. P. J., 1996. Modelling Design Exploration as Co-Evolution. Microcomputers in Civil Engineering, Issue on Evolutionary Systems in Design.
March, L, Steadman, P, 1974. The Geometry of Environment: An Introduction to Spatial Organization in Design. s.l.:M.I.T. Press.
March, Salvatore T., and Gerald F. Smith, 1995. Design and natural science research on information technology.. Decision support systems, 15(4), pp. 251-266.

Miranda, P, and Koch, D, 2013. A Computational Method For Generating Convex Maps Using the Medial Axis Transform.. Seoul, Sejong University Press.
Moudon, A. V., 1997. Urban morphology as an emerging interdisciplinary field. Urban Morphology, Volume 1, pp. 3-10.
Nagar, Atulya, and Hissam Tawfik, 2007. A Multi-Criteria Based Approach to Prototyping. Issues in Informing Science and Information Technology, Volume 4, pp. 749-756.
Neufert, Ernst, Peter Neufert, and Johannes Kister., 2012. Neufert Architec's Data. s.l., John Wiley & Sons.
Nielsen, OA, Israelsen, T & Nielsen, ER, 1997. GIS-based method for establishing the data foundation for traffic models. s.l., ESRI.
Nourian, P, Goncalves, R, Zlatanova, S, Arroyo Ahori, K, Vo, A, 2016. Voxelization Algorithms for Geospatial Applications. Methods X, Volume 3, p. forthcoming.
Nourian, P, Rezvani, S, Sariyildiz, S, van der Hoeven, F, 2015. CONFIGURBANIST: a toolkit for urban configuration analysis. Vienna, TU Wien.
Nourian, P, Sariyildiz, S, 2012. Designing for Pedestrians: A configurative approach to polycentric neighborhood design. Delft, ISUF.
Nourian, P, van der Hoeven, F, Rezvani, S, Sariyildiz, S, 2015. Easiest paths for walking and cycling:Combining syntactic and geographic analyses in studying walking and cycling mobility. London, UCL.
Nourian, P, van der Hoeven, F, Rezvani, S, 2015. Supporting Bipedalism: Geodesign for Pedestrians and Cyclists. In: Research in Urbanism Series: Geo-Design. Delft: TU Delft.
Nourian, P. Rezvani, S., Sariyildiz, S., 2013. A Syntactic Design Methodology. Seoul, Sejong University Press 2013, pp. 048:1-15.
Nourian, P. Rezvani, S., Sariyildiz, S., 2013. Designing with Space Syntax. Delft, Delft University of Technology, pp. 357-365.
Ogata, K., 1987. Modern Control Engineering. 4th, 2009 ed. s.l.:Prentice Hall.
Okasha, S., 2002. Philosophy of Science: A Very Short Introduction. s.l.: Oxford University Press, USA.
Page, Lawrence, Sergey Brin, Rajeev Motwani, and Terry Winograd., 1999. The PageRank citation ranking: bringing order to the Web., s.l.: Stanford InfoLab.
Park, H. T., 2005. Before integration: a critical review of integration measure in space syntax. Delft, TU Delft.
Peffers, K, Tuunanen, T, Rothenburger, M A, Chatterjee, S, 2007. A Design Scinece Research Methodology for Information Systems Research. Journal of Management Information Systems, 24(3), pp. 45-77.
Penn, A., Hillier, B., Banister, D. and Xu, J. , 1998. Configurational modelling of urban movement networks.. Environment and Planning B: Planning and Design , 25(1), p. 59–84.
Pigot, S., 1991. Topological models for 3d spatial information systems. s.l., ASPRS American Society of Photogrammetry and Remote Sensing, pp. 368-368.
Porta S, Crucitti P, and Latora V, 2006a. The network analysis of urban streets: a primal approach. Environment and Planning B, 33(5), pp. 705-725.
Porta, S., Crucitti P., & Latora V., 2006. The network analysis of urban streets: a dual approach. Physica A: Statistical Mechanics and its Applications, 369(2), pp. 853-866.
Portugali, J., 1999. Self-organization and the city. s.l.:Springer.
Portugali, J., 2006. The Scope of Complex Artificial Environments. In: Complex Artificial. Berlin : Springer-Verlag , pp. 9-28.
Ratti, C., 2004. Urban texture and space syntax: some inconsistencies. Environment and Planning B: Planning and Design, pp. 487-499.
Rihard Weber, James Norris, Grimmett, Stirzaker, Ross, Aldous, Fill, Grinstead and Snell, 2011. http://www.statslab.cam.ac.uk. [Online]
Available at: http://www.statslab.cam.ac.uk/~rrw1/markov/M.pdf
Rittel, H. & Webber, M., 1973. Dilemmas in a General Theory of Planning. Policy Sciences, pp. pp 155-169.
Roth, J & Hashimshony, R, 1988. Algorithms in graph theory and their use for solving problems in architectural design. computer-aided design, 20(7), pp. 373-381.
Roth, J, Hashimshony,R, Wachman, A, 1982. Turning a Graph into a Rectangular Floor Plan. Building and Environment, 17(3), pp. 163-173.
Sabudussi, G., 1966. The centrality index of a graph. Psychometrika, 31(4), pp. 581-603.
Saracevic, M., Stanimirovic, P., Masovic, S. and Bisevac, E., 2013. IMPLEMENTATION OF THE CONVEX POLYGON TRIANGULATION ALGORITHM. International Journal of Computer Mathematics.
Sariyildiz, I., 2012. Performative computational design. Konya, Selcuk University.
Scaffranek, R., 2012. SpiderWeb. Wien: TU Wien.

Schaffranek, R. and Vasku, M, 2013. SPACE SYNTAX FOR GENERATIVE DESIGN: On the application of a new tool. s.l., s.n.
Schon, D., 1987. The Reflective Practitioner. Cambridge: Cambridge University Press.
Sevtsuk, A., 2010. Path and Place: A Study of Urban Geometry and Retail Activity in Cambridge and Somerville, PhD Dissertation. Cambridge, Massachusetts, USA: Massachusetts Institute of Technology.
Shannon, C.E. and Weaver, W., 1949. The Mathematical Theory of Communication. Urbana: IL: University of Illinois Press.
Sileryte, R., 2015. Analysis of Urban Space Networks for Recreational Purposes based on Mobile Sports Tracking Application Data. s.l.:Delft University of Technology.
Simon, H. A., 1999. The Sciences of the Artificial. London: MIT Press.
Spielman, D. A., 2007. Spectral graph theory and its applications. In: s.l.:IEEE, pp. 29-38.
Spielman, D. A., 2011. Laplacian Matrices of Graphs:Spectral and Electrical Theory. s.l.:Dept. Computer Science, Yale University.
Spizzirri, L., 2011. Justification and application of eigenvector centrality.Algebra in Geography: Eigenvectors of Network, s.l.: s.n.
Ståhle A., Marcus, L. and Karlström, A., 2008. Place Syntax: Geographic Accessibility with Axial Lines in GIS. Delft, s.n.
Steadman, P., 1983. Architectural Morphology: An Introduction to the Geometry of Building Plans. s.l., Taylor & Francis, p. 276.
Tobler, W., 1993. Three presentations on geographical analysis and modeling: Non-isotropic geographical modeling speculations on the geometry of geography global spatial analysis, s.l.: National Center for Geographic Information and Analysis.
Tobler, W. R., 1970. A Computer Movie Simulating Urban Growth in the Detroit Region. Economic Geography, Volume v. 46, p. 234–240.
Tompkins, J. A., White, J. A., Bozer, Y. A., 2010. Facilities Planning. ISBN 978-0-470-44404-7 ed. s.l.:Wiely.
Torrens, P. M., 2000. How land-use-transportation models work. s.l., CASA.
Turner, A., and N. Dalton, 2005. A simplified route choice model using the shortest angular path assumption. s.l., s.n.
Turner, A., 2007. "Depthmap: A Program to Perform Visibility Graph Analysis. Istanbul, s.n.
Turner, A., 2007. From axial to road-centre lines: a new representation for space syntax and a new model of route choice for transport network analysis. Environmen & Planning B, pp. 539-555.
Tutte, W. T., 1963. How to draw a graph. s.l., s.n.
Uran, Oddrun, and Ron Janssen, 2003. Why are spatial decision support systems not used? Some experiences from the Netherlands.. Computers, Environment and Urban Systems, pp. 511-526.
Volchenkov, D, Blanchard, P, 2007. Random walks along the streets and canals in compact cities: Spectral analysis, dynamical modularity, information, and statistical mechanics. Physical Review E, 75(2).
Volchenkov, D., and Ph Blanchard., 2007. Discovering important nodes through graph entropy encoded in urban space syntax. s.l.:arXiv preprint arXiv:0709.4415.
Wei, Xuebin, and Xiaobai A. Yao., 2014. The random walk value for ranking spatial characteristics in road networks.. Geographical Analysis, 46(4), pp. 411-434.
Wildberger, N. J., 2013. Euler's triangulation of a polygon. s.l.:s.n.
Yager, R. R., 1980. ON A GENERAL CLASS OF FUZZY CONNECTIVES. Fuzzy sets and Systems, 4(3), pp. 235-242.
Zadeh, L. A., 1965. Fuzzy sets. Information and control, 8(3), pp. 338-353.
Zhang, T. Y., and Ching Y. Suen., 1984. A fast parallel algorithm for thinning digital patterns.. Communications of the ACM, 27(3), pp. 236-239.

Curriculum Vitae

BIOGRAPHY	PIROUZ NOURIAN, BORN ON 19 JULY 1981, TEHRAN, IRAN		
Academic Positions	Researcher	Design Informatics Urban Design 3D Geoinformation GIS technology	(Sep2010-Present) (Nov2015- May2016) (Oct 2014- Apr2015) (Aug 2014- Oct2014)
	Instructor	Design Informatics Geomatics	(May2011- Nov2015) (Feb 2014- Present)
@	TU Delft, Faculty of Architecture and the Built Environment		
	Dep. Architectural Engineering + Technology /Design Informatics Dep. Urbanism/Urban Design Dep. Urbanism/3D Geoinformation Dep. OTB/GIS technology		
URLs	[1] Personal Website: https://sites.google.com/site/pirouznourian/about-me [2] LinkedIn Profile: http://nl.linkedin.com/pub/pirouz-nourian/27/104/71b		
Address	Room 01+.West.010, Julianalaan, 134, 2628 BL Delft		

TEL +31(6)83846119
✉ p.nourian@tudelft.nl

Last updated:
February 21, 2016

ACADEMIC DEGREES		
2010-2016	Ph.D. in Design Informatics	TU Delft/Architecture and the Built Environment, NL
2005-2009	M.Sc. in Architecture	Tehran University of Art/Architecture & Urban Planning, IR
1999-2004	B.Sc. in Control Engineering	K.N.Toosi University of Technology/Electrical Engineering, IR

DESIGN & SPATIAL ANALYSIS TOOLS	
2012 onward	**CONFIGURBANIST:** A computational tool suite for analyzing/designing urban configurations, available for academic use at: https://sites.google.com/site/pirouznourian/configurbanist http://www.grasshopper3d.com/group/cheetah
2013 onward	**SYNTACTIC:** A computational tool suite for analyzing/designing architectural configurations, available for academic use at: https://sites.google.com/site/pirouznourian/syntactic-design http://www.grasshopper3d.com/group/space-syntax

D MODELLING TOOLS	
2014	**RASTERWORKS.DLL** A library of methods for voxel and 3D raster model generation from point clouds, curves and surfaces: https://github.com/Pirouz-Nourian
2014	**TOIDAR:** an educational toolkit for reconstruction of building 3D models from LiDAR point clouds: https://github.com/Pirouz-Nourian

TEACHING EXPERIENCES	
2016	Instructor of **Computational Design** in (MSc 2) MEGA (High-rise Design Studio), (AR0026), 2015-2016 Q4, TU Delft, Faculty of Architecture and Built Environment (BK). Instructor of **Computational Modelling** in (MSc 2) 3D Modelling, **MSc Geomatics**, (GEO1004), Directed by Dr. Sisi Zlatanova, 2015-2016 Q3, TU Delft, Faculty of Architecture and Built Environment (BK).
2015	Instructor of **Computational Modeling** in (MSc 2) 3D Modeling, **MSc Geomatics**, (GEO1004), Directed by Dr. Sisi Zlatanova, 2014-2015 Q3, TU Delft, Faculty of Architecture and Built Environment (BK). Instructor of **Computational Design** in (MSc 2) BIG and TALL Workshop, XXL buildings, **MSc Architectural Engineering** (AR0026), 2014-2015 Q3, TU Delft, BK.
2014	Instructor of **Computational Modelling** in (MSc 2) 3D Modeling, **MSc Geomatics**, (GEO1004), Directed by Dr. Sisi Zlatanova, 2013-2014 Q3, TU Delft. Instructor of **Computational Design** in (MSc 2) BIG and TALL Workshop, XXL buildings, **MSc Architectural Engineering** (AR0026), 2013-2014 Q3, TU Delft, BK.
2013	Instructor of **Computational Design** in (MSc 2) BIG and TALL Workshop, High-Rise buildings, **MSc Architectural Engineering** (AR0026), 2013-2014 Q1, TU Delft, BK. Instructor of **Computational Design** in (MSc 2) XXL Design Studio (AR0025), 2012-2013 Q3, TU Delft, BK.
2012	Instructor of **Computational Design** in MSc 3 **Computational Architecture** (AR4AC010), 2011-2012 Q3, TU Delft, BK. Instructor of NURBS-CAD **Parametric Design** in (MSc 2) XXL Design Studio (AR0025), 2011-2012 Q3, TU Delft, BK. Instructor of **Parametric Design** in(MSc 2) Bucky Lab Design CAD (AR1AE015), 2011-2012 Q3 & Q1, TU Delft, BK.
2011	Instructor of **Parametric Design** in (MSc 2) XXL Design Studio (AR0025), 2010-2011 Q3, TU Delft, BK. Instructor of **Parametric Design** in (MSc 2) Bucky Lab Design CAD (AR1AE015), 2011-2012 Q3 & Q1, TU Delft, BK.
2010	Instructor of **Parametric Design** in(MSc2) Bucky Lab Design CAD (AR1AE015), 2010-2011 Q3 & Q1, TU Delft, BK.
2009	Assistant Lecturer of (MArch 3) **Design Methods and Techniques**, at Tehran University of Art, fall 2009 Tutor of **Computer Aided Architectural Design** for graduate students in University of Art, Modelling in Rhinoceros and its plugins, fall 2009
2008	Teaching Associate of **Architectural Design Studio** (series of main courses of BA in Architecture) at the University of Art and the University of Shariati (Tehran), spring 2009, fall 2008, spring 2008
2004	Teaching Assistant in B.Sc. course **Industrial Control**, spring 2004 at KNTU
2003	Teaching Assistant in B.Sc. course **Linear Control Systems (Control Theory)**, fall 2003 at KNTU

INTERNATIONAL WORKSHOP INSTRUCTION	
2015	*Cityscape Configuration* @ eCAADe 2015, @ **TU Wien**, together with Philip Belesky, Vienna, September 2015
2014	*Generative Syntax in Architecture and Urban Design* @ **UCL** in *AAG 2014* together with Richard Schaffranek (TU Wien)
2013	*URBAN DATASCOPE*, @ eCAADe 2013 conference, TU Delft Sep 2013 *Tarlabasi DATASCOPE* @ **Istanbul Technical University**, May 2013
2012	Lecturer & Tutor of computational urban design in *Measuring Urbanity* Seminar & Workshops at **FAUTL - Faculty of Architecture, TU Lisbon**, May 2012
2011	Workshop Instructor: HYPERBODY workshop on Computational Design Technology and Methodology, MSc3 & MSc1 Studios, Q1, September 2011, at TU Delft

THESIS SUPERVISION	
2015	MSc Geomatics Rusne Sileryte, Cum Laude @ **TU Delft,** Graduation Professor: Dr. Stephan van der Spek, Daily Supervisor: Ir. Pirouz Nourian, third mentor: Dr. Hugo Ledoux**:** *Analysis of Urban Space Networks for Recreational Purposes based on Mobile Sports Tracking Application Data.* MSc Geomatics Marco Lam@ TU Delft, Graduation Professor: Dr. Jantien Stoter, Daily Supervisor: Ravi Peters, third mentor: Ir. Pirouz Nourian: *Creating the Medial Axis Transform for billions of LiDAR points using a memory efficient method***.** MSc Geomatics Damien Mulders @ **TU Delft,** Graduation Professor: Dr. Jantien Stoter, Daily Supervisor: Dr. Hugo Ledoux, third mentor: Ir. Pirouz Nourian**:** *Automatic repair of geometrically invalid 3D City Building models using a voxel-based repair method.* MSc Geomatics Kaixuan Zhou @ **TU Delft,** Graduation Professor: Dr. Sisi Zlatanova, Daily Supervisor: Dr. Ben Gorte, third mentor: Ir. Pirouz Nourian**:** *Exploring Regularities for Improving Quality of Facade Reconstruction from Point Cloud.*
2014	MSc Geomatics Eva van der Laan @ **TU Delft**, Graduation Professor: Dr. Stephan van der Spek, Daily Supervisor: Ir. Wilko Quak, third mentor: Ir. Pirouz Nourian**:** *An indoor positioning method using Bluetooth Low Energy technology* MSc Geomatics Xu Weilin @ **TU Delft**, , Graduation Professor: Dr. Sisi Zlatanova, Daily Supervisor: Ir. Liu Liu, Third mentor/Reader Ir. Pirouz Nourian: *An indoor positioning method using WiFi routers*
2013	MSc Architecture Samaneh Rezvani @ **Politecnico di Milano**, Graduation Professor: Dr. Andrea Rolando, Daily Supervisor: Ir. Pirouz Nourian:, *An interactive computational methodology for plan lay out using Space Syntax*

COMMITTEE MEMBERSHIP	
2016	Jury Member of The 11th **European and Regional Planning Awards of ECTP-CEU**, (European Council of Spatial Planners - Conseil Européen des Urbanistes) Bruxelles Member of Review Committee 7th **SimAUD Symposium** on Simulation for Architecture and Urban Design, UCL, London Member of Review Committee 34th **eCAADe Conference**, Oulu, Finland
2015	Member of Review Committee 10th **Space Syntax Symposium**, UCL, London Member of Review Committee 33rd **eCAADe Conference**, TU Wien, Vienna

RELEVANT PUBLICATIONS	
2016	*Nourian, P*, Goncalves, R, Zlatanova, S, Arroyo Ahori, Vo, A.V., (2016) **Voxelization Algorithms for Geospatial Applications**, MethodsX, Elsevier [*URL*] Zlatanova, S, *Nourian, P*, Goncalves, R, Vo, A.V., (2016) **TOWARDS 3D RASTER GIS: ON DEVELOPING A RASTER ENGINE FOR SPATIAL DBMS,** proceedings of ISPRS WG IV/2 Workshop "Global Geospatial Information and High Resolution Global Land Cover/Land Use Mapping", April 21, 2016, Novosibirsk, Russian Federation, [*URL*] Sileryte, R, *Nourian, P*, (2016) **Modelling Spatial Patterns of Outdoor Physical Activities using Mobile Sports Tracking Application Data**, Springer Lecture Notes on Geoinformation and Cartography, Accepted. *Nourian, P*, van der Hoeven, F, Rezvani, S, Sariyildiz, S, (2016) **Supporting Bipedalism: Computational Analysis of Walking and Cycling Accessibility for Geodesign Workflows**, RIUS Research in Urbanism Series, GEODESIGN, TU Delft, Accepted. *Nourian, P*, van der Hoeven, F, Sariyildiz, S, Rezvani, S, (2016) **Spectral Modelling of Spatial Networks**, SimAUD, UCL, London, ACM press, Accepted.

>>>

RELEVANT PUBLICATIONS	
2015	*Nourian, P.*, Rezvani, S., Sariyildiz, S, van der Hoeven, F. (2015). **CONFIGURBANIST - Urban Configuration Analysis for Walking and Cycling via Easiest Paths**, proceedings of the 33rd eCAADe Conference, TU Wien, Vienna [*URL*] *Nourian, P.*, van der Hoeven, F, Rezvani, S., Sariyildiz, S. (2015). **Easiest paths for walking and cycling: Combining syntactic and geographic analyses in studying walking and cycling mobility**, proceedings of the 10th Space Syntax Symposium, UCL, London [*URL*]
2014	Goncalves, R, Ivanova, M, Kersten, M, Scholten, H, Zlatanova, S , Alvanaki, F, *Nourian, P* & Dias, E (2014, November 3). **Big Data analytics in the Geo-Spatial Domain**. Groningen, Big Data Across Disciplines: In Search of Symbiosis, conference 3-5 November 2014. [*URL*] Chen, J, Sileryte, R, Zhou, K, *Nourian, P* & Zlatanova, S (2014). **Automated 3D reconstruction of buildings out of point clouds obtained from panoramic images**. Walnut Creek, USA: CycloMedia. (TUD)
2013	*Nourian, P.* Rezvani, S., Sariyildiz, S. (2013). **Designing with Space Syntax.** Proceedings of eCAADe 2013, (pp. 357-366). Delft. *Nourian, P.*, Rezvani, S., Sariyildiz, S. (2013). **A Syntactic Design Methodology**. Proceedings of 9th Space Syntax Symposium. Seoul. *Nourian, P.*, Sariyildiz, S., Rezvani, S. (2013). **An interactive computational methodology for urban mixed-use allocation according to density distribution, network analysis and geographic attractions**. Proceedings of Changing Cities Conference. Skiathos
2012	*Nourian,P*,Sariyildiz,S, **A Computational Walkability Assessment Model**, CIB Webinars on measuring Urban Sustainability [1] *Nourian,P*,Sariyildiz,S, **A configurative approach to neighborhood planning and design, promoting pedestrian mobility: An interactive design method for polycentric distribution of built space according to walkability, attractions and topographic features**. Proceedings of New Configurations ISUF International Conference on Urban Morphology (due to be published by March 2013) *Nourian, P*, **On Computational Design Methodology**, CIAUD: Research Centre for Architecture, Urban Planning and Design, Measuring Urbanity Seminar, Proceedings of International Seminar & Workshop, Lisbon, May 7th-12th, 2012, FAUTL - Faculty of Architecture, Technical University of Lisbon (due to be published by April 2013)
2011	Beirão, J, *Nourian, P*, Mashhoodi, B, 2010, **A Parametric Urban Design System**, in eCAADe Slovenia Conference Proceedings, presented in September 2011, Ljubljana, Slovenia. Beirão, J, *Nourian, P*, Van Walderveen, B, 2010**, An Integrated Process of Urban Pattern Generation and Route Structure Analysis**, in IASDR 2011 proceedings, presented in October 2011, Delft, the Netherlands.

PROFESSIONAL DESIGN EXPERIENCE	
2006-2010	Architectural Designer & Design Technical Manager @ Gashtaar, Tehran, Iran *http://tractureoffice.com/*
2005-2007	Freelance Architect & Interior Designer in three projects of interior design-build for residential and commercial places in Tehran

PROFESSIONAL SKILLS	
Methods	Mathematical Modelling, Scientific Computing, Computational Modelling & Simulation, 3D, Spatial Analysis and Synthesis, Design Methodology, Architectural Design, Urban Design
Software	C# (advanced), VB.NET (advanced), Visual Studio IDE, Python, Grasshopper3D® (advanced), Rhinoceros-3D® (advanced), AutoCAD, Depth Map, Microsoft Office, MATLAB, Mathematica