Coloured–Edge Graph Approach for the Modelling of Multimodal Networks

Felipe Eduardo Lillo Viedma

A thesis submitted to
Auckland University of Technology
in fulfillment of the requirements for the degree of
Doctor of Philosophy (PhD)

2011

School of Computing and Mathematical Sciences
Primary Supervisor: Dr. Andrew Ensor

Contents

List of Figures

List of Tables

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Auckland, 2011

Felipe Eduardo Lillo Viedma

Acknowledgements

I firstly would like to thank my supervisor Andrew Ensor for all his patience and valuable help in developing this thesis. Undoubtedly, this research could not have been finished without his important guidance. Additionally, I want to personally thank Andrew for opening me the door to new research subjects most of which will certainly influence my future as an academic.

I also would like to acknowledge Católica del Maule University (Chile) for giving me the opportunity and financial support for undertaking my doctoral studies. It is my desire to return all the scientific experience accumulated these four years to this university (where I have worked for ten years) and become a real contribution in its scientific development.

I want to express my gratitude to my PhD mates Frederik, Anuj and Waseem for their friendship and help in the last months of my studies.

I finally wish to thank most–of–all my wife Lorena. For her unconditional support and company all these years. This work is in great extent a result of her devotion, love and bravery in facing new challenges, qualities that I have been able to strengthen in myself thanks to her.

Abstract

Many networked systems involve multiple modes of transport. Such systems are called multimodal, and examples include logistic and telecommunication networks, biomedical phenomena, conflict resolution models and manufacturing processes. Existing techniques for determining minimal paths in multimodal networks have required either heuristics or else application-specific constraints to obtain tractable problems, removing the multimodal traits of the network during analysis. In this thesis weighted coloured–edge graphs are introduced to model multimodal networks, where colours represent the modes of transportation. Optimal paths are selected using a partial order that compares the total weights in each colour, resulting in a Pareto optimal set of shortest paths. The cardinality of this set is at the core of the model's tractability.

Tractability and applicability of the coloured–edge graph are addressed in this work. The tractability is firstly studied experimentally by using random as well as pathological instances of the colored–edge graph. Next, upper bounds on the cardinality of the Pareto set are established. An upper bound which is exponential in k (number of colours) is first presented. Subsequently, a probabilistic bound is developed for bicoloured–edge graphs whose weights are randomly drawn according to a bounded probability density function. An $O(n^3)$ bound on the expected number of minimal paths (where n is the number vertices) is established.

The applicability of the approach is studied by means of data obtained from real multimodal transportation networks. Three cases are studied. Case (1) is a comparative analysis based on the multimodal

transportation systems of New Zealand and Europe. The data used in the construction of the networks consist of digitized maps obtained from official GIS libraries. Case (2) is a large multimodal network that reproduces transport options in France. This instance is mainly focused on assessing the performance of the coloured–edge graph for very large networks. Finally, Case (3) utilizes air traffic information to build a multimodal network with a large number of modes. Modes in this case correspond to different international airlines. An important aspect of this practical study is that the multimodal networks are larger than most of those previously analyzed in the literature.

The coloured–edge graph is shown in this research to be typically tractable without the need to apply any application–specific heuristic or constraints. This provides a new perspective in the analysis and optimization of systems that can be modelled as multimodal networks.

Chapter 1

Introduction

1.1 Context and Background

1.1.1 Multimodal networks

Networks play an important role in our lives. Power networks provide electricity to homes. Telephone and computer networks allow us to communicate and interchange information at local and global levels. Motorway systems, rail and airline networks make it possible to transport products. Manufacturing systems turn raw materials into products by different sequences or machines. One common features in all these domains is the determination of an optimal pathway linking at least two points in a network. The determination of a shortest path is a problem that has been addressed by several fields of inquiry, including applied mathematics, computer science, engineering, management, and operations research. An abundant and long tradition characterizes this problem, tracing its origins to the 19th century, and reaching an important peak in 1959 thanks to Dijkstra's algorithm.

During the last three decades, one particular case has drawn considerable attention in network optimization research; this is the optimization of multimodal transportation systems which are networked systems involving multiple modes of transport. Figures 1.1 and 1.2 portray two different applications of multimodal networks: a biological system presented by Heath & Sioson [1] in which the role

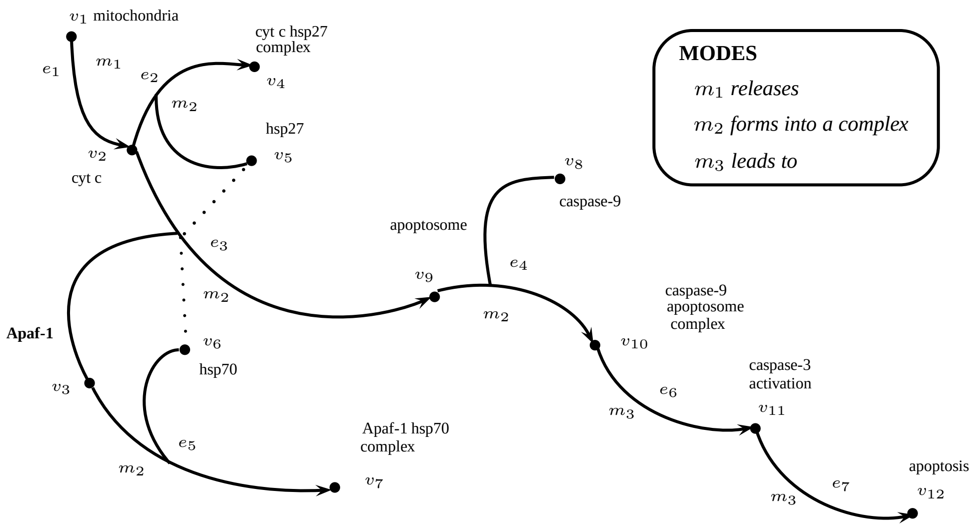


Figure 1.1: A multimodal network from Biology. Source [1].

of heat shock proteins in apoptosis is modelled as as a network with three modes, and a multimodal network in the context of the conflict resolution problem from Kilgour et al. [2] where each arc is labeled with a different decision maker (UR, MoE and LG).

Attention towards multimodal networks has mainly arisen as a consequence of the logistic boom phenomenon which can be accounted by two main factors: globalization, and the quest for new competitive advantages inside companies. Two proofs of these factors stem from reports [3] and [4] elaborated in The European Union and USA, respectively. Both documents establish the importance of multimodalism as a competitive element and the need to address this issue from a scientific outlook.

In essence, a multimodal network is a generalization of single mode network; and so relevant to several research fields such as operations research, computer science, applied mathematics, management and logistics. Ahuja et al. [5] assert that many applications in these fields not only happen "naturally" in some physical networks, but also in situations that apparently are quite unrelated to networks. Furthermore, because networked systems arise in so many problem contexts, applications are scattered throughout different literature fields. Areas such as computer networks, biomedicine and manufacturing have begun to utilize

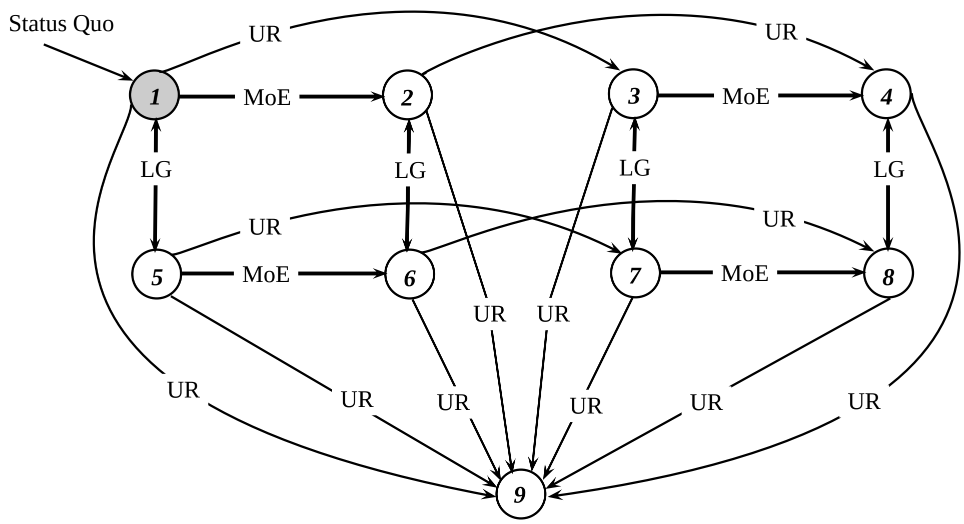


Figure 1.2: A multimodal representation of a Conflict Resolution Problem. Source Kilgour et al. [2].

multimodal network models. For instance, Abrach et al. [6] design a computer application for the development of sensor wireless networks. The architecture of the system is based on a multimodal network that uses several threads as modes. In biomedicine, Chen et al. [7] introduce modelling techniques for some biological processes based on multimodal networks. Heath & Sioson in papers [8] and [9] provide a mathematical context for modelling biological systems by multimodal networks. Madeiros et al. [10] and Kiesmüller [11] are two authors dealing with manufacturing processes and their multimodal representation. In the paper by Nigay & Coutaz [12] an analysis of a multimodal communication system is presented.

Scientific literature in multimodal networks has been mainly focused on analyzing practical logistic applications for freight or urban transportation. References that discuss views about these application fields include Boyce [13] and Macharis & Bontekoning [14]. Consequently, the richness and variety of applications that multimodal networks deliver is not immediately evident to many network designer communities. Nonetheless, as a system in which several means of transport are available, a multimodal network is able to represent a wide spectrum of real life phenomena far beyond the field of logistics.

1.1.2 Modelling techniques for multimodal networks

To model multimodal networks, researchers have used a significant range of operations research techniques.They can be classified into three predominant domains: mathematical programming, weighted graphs and multi–weighted graphs.

Mathematical programming

Mathematical programming is a technique characterized by making use of linear or non–linear formulations for representing a multimodal network by a set of equations.

Hillier & Lieberman [15] stipulate that linear programming techniques are suitable when each decision variable is a linear combination of the problem parameters. Integer programming and mixed integer programming stand out as the most common linear programming techniques used for multimodal modelling. Sample papers using linear programming as a modelling tool are [16] and [17].

Non-linear programming is another renowned modelling technique for multimodal networks. It is mainly used to build intricate cost functions, and principally deals with second order equations satisfying convex or concave properties. Examples are found in papers [18], [19] and [20] where non-linear programming is used as the main modelling approach.

In the mathematical programming approach, mode options are visualized as decision variable indices, which considerably increases the complexity of the problem. Relaxation or cutting plane techniques are commonly used to make the problem tractable. Interesting papers tackling general views of mathematical programming for intermodal transportation (the transportation of goods) and urban transportation are [21] and [22] respectively.

The weighted graph approach

In the weighted graph approach, a node typically represents a location, such as a warehouse, transportation hub or network router, and an edge represents a transportation link, such as rail line, a bus or a wireless connection. A variety of graphs have been used to study these transport systems, such as digraphs, multigraphs, hypergraphs and grid graphs. Ayed et al. [23] provide a general classification for multimodal network models based on weighted graph approaches. In particular, their article emphasizes the use of multigraphs, in which there might be multiple edges between two nodes, and the use of grids in which a grid is overlayed on a planimetric map. Both can result in dense graphs, which require edge reduction techniques to make their analysis tractable. In practice such reductions rely on enforcing constraints on feasible edges in order to build a specific path. Papers making use of such graphs are [24], [25], [26] and [27]. Hypergraphs are another type of graph used in some articles. Lawler [28] defines a hypergraph as a generalization of a graph, where edges can connect any number of vertices. In the multimodal context, such graphs have found interesting applications in biology and urban transportation. Sample papers using hypergraphs to represent multimodal networks are [8], [9] and [29].

In effect, the weighted graph approach only utilizes mode information during the application of constraints, removing the multimodal traits from the network during analysis. The analysis in this approach is very application-dependent as it relies on applying application-specific constraints.

The multi–weighted graph approach

Multi–weighted graphs have been extensively utilized for the multicriteria shortest path problem which has become a fruitful branch of research since the 1980s. Tarapata [30] and Soroush [31] provide reviews about this topic. Basically, the approach assigns multiple weights to each edge. In particular, the bicriteria shortest path problem assigns two weights to each edge, such as cost and time.

Optimality in the multi–weighted graph approach is commonly established by the use of a partial order relation which results in a Pareto optimal set of paths that are candidates for the sought shortest path. There is little literature that directly applies multi–weighted graphs for modelling multimodal networks, but the goal of the multicriteria shortest path problem is essentially the same as for the shortest path problem in multimodal networks. Although some articles developing formulations for multimodal networks based on the multicriteria shortest path problem can be identified, they preferentially use partial orders to compute optimal paths by cost and time, leaving the mode options as an outcome of the optimal route. Multimodal network models whose mainstay is a multi–weighted graph can be found in papers [32], [33] and [34]. The multicriteria shortest path path problem has been proved to be intractable. Its complexity was shown by Hansen [35] to be exponential in the worst case. However, Loui [36] later pointed out that Pareto sets for some graphs with multidimensional weights have polynomial average case cardinalities. Finally, Muller-Hannemann & Weihe [37] also studied the behavior of this problem experimentally. They stated that the tractability of the multicriteria shortest path problem is inextricably connected with the cardinality of the Pareto set.

The multi–weighted graph approach requires the application of constraints during analysis to make the problem tractable. As a result, the final Pareto set obtains a manageable cardinality.

1.2 Research Framework

After several years of scientific study, multimodal network optimization is still a fruitful research area. The potential is tremendous because of its ability to represent a variety of real life transportation phenomena in which more than one mode operates. Nevertheless, there has been a lack of research into modelling techniques and mathematical methods specifically developed for multimodal networks. Practical applications in the specific fields of freight and urban transportation have been the main research focus all these years rather than fundamental research. As a consequence, most of the modelling approaches used in specific applications cannot be generalized and require the use of application–specific constraints or heuristics to be efficient.

A modelling approach based on a coloured–edge graph is introduced in this thesis. The approach is promoted as a general tool for the study of multimodal networks. Moreover, the approach is capable of overcoming some drawbacks of previous techniques by avoiding the need for either reduction techniques or application–specific constraints. Although an intuitive approach to modelling networks in which there are multiple transportation modes, it does appear to give a new perspective and truly general approach for multimodal networks.

The coloured–edge graph is a graph modelling approach based on the labelling of edges with colours to represent available transportation modes and paths use the summed weight separately in each mode. By labelling edges in this manner, the multimodal nature of the network is maintained during an optimization analysis. This is an important model feature, since current modelling techniques require the removal of modes from the analysis to make the problem tractable. The coloured–edge graph approach is formally described in Chapter 2.

Shortest path problems lie at the core of many network optimization problems. In this thesis, the single–source shortest path problem is investigated for coloured–edge graphs. Nonetheless, the shortest path problem in a colored–edge graph significantly differs from traditional formulations since minimal path weights are partially ordered instead of linearly ordered. Consequently, the shortest path is not limited to just one specific path, rather a set of paths that are candidates for the optimal answer. These optimal paths are seen in this thesis as *Pareto efficient*, a concept that is formally introduced in Chapter 2.

To compute optimal paths in a weighted coloured–edge graph, a Dijkstra–like algorithm is presented in Chapter 3. This algorithm is based on a partially ordered data structure that is able to deal with reasonably large multimodal networks from the literature. One argument in favour of generating a set of optimal paths is the possibility to efficiently perform post–optimal analysis. By modifying parameters in a cost function, a user of the algorithm can estimate how much change is needed to replace one optimal solution in the set by another. A description about the benefits of post–optimal analysis is also provided in this Chapter.

An experimental study complements Chapter 3. The algorithm performance strongly depends on the number of optimal paths (Pareto set) so that the study of the cardinality of the Pareto set becomes crucial for estimating the feasibility of practical implementations. By generating different random instances of the coloured–edge graph, the algorithm produces sufficient data to conjecture several facts about the cardinality of the final Pareto set. Additionally, a section of this chapter experimentally compares the performance of three different data structures for the handling of minimal paths. Surprisingly, a traditional partially ordered priority queue performs efficiently.

The cardinality of the final Pareto set is at the heart of model's tractability and constitutes the central issue of Chapters 4 and 5. The cardinality of the Pareto set is shown in Chapter 4 to be bounded by k^{n-1} for coloured–edge graphs with positive arc weights. Furthermore, bounds for special instances of the coloured–edge graph are developed together with a NP–completeness analysis. A probabilistic analysis is carried out in Chapter 5 to explain the behavior of the model under random inputs. An argument based on the findings of Röglin & Vöcking [38] provides an $O(n^3)$ probabilistic bound on the expected number of minimal paths for bicoloured–edge graphs whose weights are drawn according to a bounded probability density function. The probabilistic bound on the expectation implies that bicoloured–edge graphs are typically tractable in practice. This idea is also supported by the experiments presented throughout this thesis.

Any modelling tool proves to be useful if it can be successfully applied in practice. Real transportation networks are the main ingredient of Chapter 6 to demonstrate the level of applicability of the proposed approach. Large multimodal networks are built by taking transportation datasets from several countries and world–wide airline data. The approach effectively deals with such networks bolstering the applicability of the approach.

1.3 Bibliographic Notes

Some results of this thesis have been published elsewhere. The study performed in New Zealand and European multimodal system (Chapter 6) was published by Lillo & Schmidt in the Proceedings of the 45th Annual Conference of the Operation Research Society of New Zealand [39]. The edge–coloured graph approach as well as the experimental results of Chapter 2 were recently submitted by Lillo & Ensor to be peer–reviewed [40]. Moreover, both the algorithm for determining Pareto minimal paths and the post–optimal analysis of the Pareto set described in Chapter 3 are also part of this paper. The coloured–edge graph as a modelling tool for multimodal networks was presented by Lillo in EURO conference 2009 [41]. The upper bound on the number of minimal paths introduced in Chapter 4 was presented by Ensor in EURO 2010 conference [42]. Likewise, results based on the probabilistic bounds were also presented by Lillo in this conference [43]. Finally, the main findings in Chapters 4 and 5 will be part of a paper currently under development by Lillo & Ensor.

Chapter 2

The Coloured–Edge Graph

In this chapter the coloured–edge graph is introduced. Firstly, a short scientific review is carried out. It is noticed that previous work with these graphs is very scarce. The review found only three papers directly addressing the concept of coloured–edge graph. Moreover, practical applications are scarce throughout the literature. Next, the coloured–edge graph and some special structures thereof are formally defined. Finally, the coloured–edge graph approach is presented as a modelling tool for multiobjective combinatorial problems and multimodal networks. For the former, a useful graph transformation allows a multiobjective combinatorial problem to be turned into a coloured–edge graph. As a result, some multiobjective combinatorial problems can be optimized by using its graphical representation as a weighted coloured–edge graph.

2.1 Introduction

A well–known problem in graph theory involving colours and graphs is the graph coloring problem. This problem is about the optimal assignment of colours (labels) to elements of a graph according to certain constraints. Vertex colouring is one variant of this problem. This problem asks for a way to colour vertices in a graph using the least number of colours such that no two adjacent vertices share the same colour. Malaguti & Toth [44] provide a survey of the vertex colouring problem including algorithms and generalizations. Analogously, the edge colouring problem looks for an assignation of colours to edges using the least number of colours so that no two incident edges have the same colour.

Extensive literature has been devoted to the study of the graph colouring problem. Du & Pardalos [45] provide a bibliographic survey depicting the major findings in connection with this problem. Many approaches and algorithms have been engineered for dealing with graph colouring problems in both graph theory and computer science. However, the coloured–edge graph conceived in this thesis significantly differs from the graph colouring problem in how colours are treated. The coloured–edge graph is not related to how colours are assigned to elements of a graph, rather the problem is to efficiently compute an optimal structure of a given coloured–edge graph such as a minimum spanning tree or a shortest path.

Unlike the graph colouring problem, optimization problems that utilize models similar to weighted coloured–edge graphs have received little attention in the literature. Clímaco et al. [46] experimentally studied the number of spanning trees in a weighted graph whose edges are labelled with a colour. In that work, weight and colour are two criteria both to be minimized and the proposed algorithm generates a set of non–dominated spanning trees. The computation of coloured paths in a weighted coloured–edge graph is investigated by Xu et al. [47]. The main feature of their approach is a graph reduction technique based on a priority rule. This rule basically transforms a weighted coloured–edge multidigraph into a coloured–vertex digraph by applying algebraic operations to the adjacency matrix. Additionally, the authors provide an algorithm to identify coloured source–destination paths. Nevertheless, the algorithm is not intended for general instances because its input is a unit–weighted coloured multidigraph and only paths not having consecutive edges equally coloured are considered. Finally, Manoussakis [48] studied the computation of paths with specific colour patterns in unweighted coloured–edge graphs. Particularly, his study focuses on alternating coloured–edge paths in complete coloured–edge graphs. Algorithms for finding alternating coloured–edge paths are also presented.

The optimization of weighted coloured–edge graphs is analyzed in this thesis. Particularly, the single–source shortest path problem in general weighted coloured–edge multigraphs is researched. Its capacity for accurately describing multimodal networks is assessed and described.

2.2 Formal Definition

A coloured–edge graph uses colours to represent transport options in a network. In the context of multimodal networks, such options are available transport links between locations. Vertices are employed to represent locations which in practice can be road junctions, warehouses, hubs, bus stops or suchlike. According to the system under representation, either directed or undirected edges can be considered. Likewise, weighted or unweighted instances can be part of the model. However, in most of the material in this thesis, it is assumed that the underlying coloured–edge graph is directed and weighted.

Definition 2.2.1. A *weighted coloured-edge graph* $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ consists of a directed multigraph $\langle V, E \rangle$ with vertex set V and edge set E, a *weight* function $\omega\colon E \to \mathbb{R}^+$, and a (surjective) *colour* function $\lambda\colon E \to M$, where M is a set of possible colours for the edges.

The graph $\mathbf{G}$ is said to be *finite* if both V and E are finite sets, in which case M is also finite with cardinality k. In terms of multimodal networks, the finiteness of M is justified since the number of available modes to connect locations is always finite. Figure 2.1 yields an illustration of a directed coloured–edge graph with

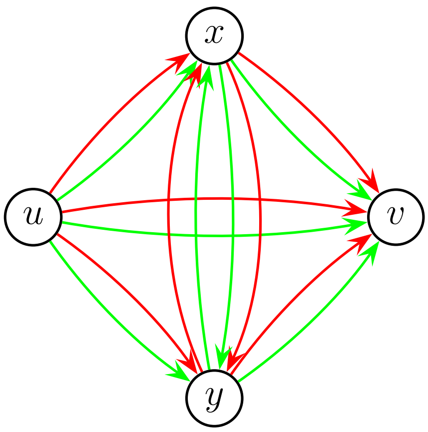


Figure 2.1: A directed coloured–edge graph.

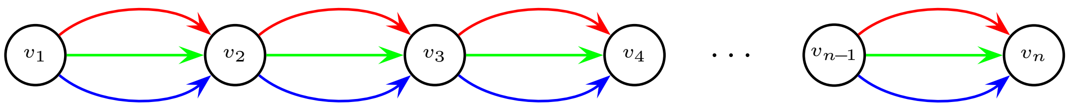


Figure 2.2: A coloured–edge chain.

$M = \{red, green\}$, where the edge weights have been omitted.

Note that there is no restriction placed on the number of edges e_{uv} from a vertex u to a vertex v. However, as this thesis investigates shortest paths and all weights are positive, it is presumed that graphs do not have any self-loops. Similarly, it can be presumed that for any two vertices $u, v \in V$ and colour $c \in M$ there is at most one edge $e_{uv} \in E$ from u to v for which $\lambda(e_{uv}) = c$. Therefore, for a finite coloured–edge graph with $n = |V|$, $m = |E|$, $k = |M|$, there is a bound on the number of edges, given by $m \leq kn(n-1)$.

One special graph that will be frequently cited throughout this thesis is the coloured–edge chain graph.

Definition 2.2.2. A weighted coloured–edge graph $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ is called a chain if the vertex set V can be ordered $1, 2, \ldots, n-1, n$, where $n = |V|$, and the graph only has edges $e_{uv} \in E$ for $v = u + 1$.

A coloured–edge chain will prove to be crucial in analyzing the tractability of the coloured–edge graph approach. A corresponding example is shown by Figure 2.2 for $k = 3$ (again with weights omitted).

2.3 Paths and Path Weights

Paths establish connectivity patterns in graphs. In this section, the definitions of path, path length and path weight for the coloured–edge graph are given.

Definition 2.3.1. Let u and v be two given vertices of $\mathbf{G}$. A coloured–edge path p_{uv} is a sequence of edges of the form $\{e_{x_0x_1}, e_{x_1x_2}, e_{x_2x_3}, \ldots, e_{x_{l-1}x_l}\}$, joining vertices $u = x_0$ and $v = x_l$, where each $x_i \in V$. The path is called simple if the vertices $x_0, x_1, \ldots, x_l$ are all distinct.

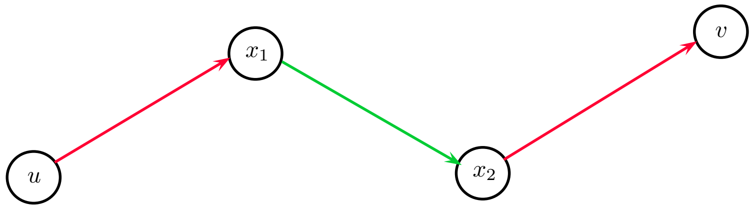


Figure 2.3: A coloured–edge path from u to v.

Figure 2.3 depicts a simple coloured–edge path with three edges from u to v. Two nodes in the coloured–edge graph are said to be connected if there exists a path from one of the vertices to the other. Next definitions establish the difference between path length and path weight.

Definition 2.3.2. The length of a coloured–edge path p_{uv} is the number of edges that the path uses. The length can be zero for the case of the empty path p_{uu} from a single vertex u to itself.

Definition 2.3.3. For any colour $i \in M$ and for any path p_{uv} between two vertices u and v, the path weight $\omega_c(p_{uv})$ in colour c is defined as,

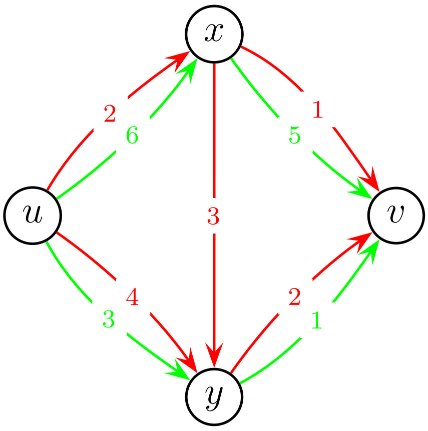
$$\omega_c(p_{uv}) = \sum_{e_{x_i x_{i+1}} \in p,\ \lambda(e_{x_i x_{i+1}})=c} \omega(e_{x_i x_{i+1}}).$$

So defined, the total path weight is represented as a k-tuple $(\omega_{c_1}(p_{uv}), \ldots, \omega_{c_k}(p_{uv}))$ giving the total weight of the path in each colour. As an example, assume all edges weight in Figure 2.3 are one. Hence, the weight tuple for this path is $(2, 1)$ by taking red as the first tuple component.

2.4 Pareto Set of Minimal Paths

From a computation standpoint, the goal is to determine paths in a coloured–edge graph whose weights satisfy a specific criterion. Such a criterion is established by a preference relation on path weights.

Definition 2.4.1. Let $\mathcal{P}_{uv}$ be the set of all paths from u (source) to v (destination) in G. A binary relation between two paths p_{uv} and p'_{uv} in $\mathcal{P}_{uv}$, is defined by $p_{uv} \leq p'_{uv}$ if and only if $\omega_c(p_{uv}) \leq \omega_c(p'_{uv})$ for all c.



Set of u–v paths		
Path	Weight Tuple	Minimal
u-x-v	$(3, 0)$	Yes
u-x-v	$(0, 11)$	No
u-x-v	$(2, 5)$	No
u-x-v	$(1, 6)$	No
u-y-v	$(6, 0)$	No
u-y-v	$(0, 4)$	Yes
u-y-v	$(4, 1)$	No
u-y-v	$(2, 3)$	Yes
u-x-y-v	$(7, 0)$	No
u-x-y-v	$(5, 1)$	No
u-x-y-v	$(5, 6)$	No
u-x-y-v	$(3, 7)$	No

Figure 2.4: Small weighted coloured–edge graph and its corresponding $\mathcal{P}_{uv}$ set.

The relation $\leq$ is clearly reflexive and transitive and gives a partial order on the k-tuple path weights, but only a preorder on the paths themselves as multiple paths might have the same total path weight.

The imposition of a preference relation on $\mathcal{P}_{uv}$ produces a subset composed of just minimal (or shortest) paths. All tuples in this set come with a special property termed *Pareto optimality*.

Definition 2.4.2. The Pareto set of minimal paths, $\mathcal{M}_{uv}$, is a set of paths joining two vertices u and v in a weighted coloured–edge graph such that $\mathcal{M}_{uv} = \{p_{uv} \in \mathcal{P}_{uv} \mid \forall p'_{uv} \in \mathcal{P}_{uv} \text{ with } \omega(p'_{uv}) \neq \omega(p_{uv}), \exists \text{ colour } c \text{ such that } \omega_c(p_{uv}) < \omega_c(p'_{uv})\}$.

This set has an important characteristic: for any $p_{uv} \in \mathcal{M}_{uv}$, it is impossible to determine a path p'_{uv} from u to v which has smaller weight than p_{uv} in some of its k colours without at least one of the other weights being larger, analogously to Martins [49]. Figure 2.4 illustrates sets $\mathcal{P}_{uv}$ and $\mathcal{M}_{uv}$ for an example of a small weighted coloured–edge graph. In this figure the third column of the table identifies minimal paths. Note that $|\mathcal{P}_{uv}| = 12$ and $|\mathcal{M}_{uv}| = 3$.

The definition of adjacent minimal paths is now introduced.

Definition 2.4.3. Two distinct paths p and p' are called adjacent if there is no path p'' that lies inside the axis aligned box in $\mathbb{R}^k$ that has $\omega(p)$ and $\omega(p')$ at opposite corners. More precisely, there is no path p'' for which $\min\{\omega_c(p), \omega_c(p')\} < \omega_c(p'') < \max\{\omega_c(p), \omega_c(p')\}$ for every colour c.

For convenience, variables q and ℓ are introduced as the cardinalities of $\mathcal{M}_{uv}$ and $\mathcal{P}_{uv}$ respectively.

2.5 Multiobjective Combinatorial Optimization Problems

The classical multiobjective optimization problem should not be confused with its combinatorial version. The difference between them lies in the feasible set described by their constraints. For the multiobjective combinatorial optimization problem, this set is strictly bounded and finite, Ehrgott [50]. Decision variables are binary and usually used in practice for representing yes/no decisions.

Suppose X is a set and $f_1, \ldots, f_r : X \to \mathbb{R}$ are real valued functions. Multiobjective optimization is concerned with finding a value of $x \in X$ for which $((f_1(x), f_2(x), \ldots, f_r(x))$ is considered Pareto optimal, optionally subject to constraints. In multiobjective combinatorial optimization the set X is finite.

For example, a multiobjective combinatorial optimization problem might be to minimize the total weight and number of edges in a path between two vertices for a weighted graph. Let A be the set of all edges between two vertices in the graph and let $X \subseteq 2^A$ be the subset of the power set of A that consists of all paths between two vertices in the graph. Take $f_1(x)$ to be the sum of all edge weights in the path and $f_2(x)$ to be the number of edges $|x|$ in the path x. Then the multiobjective combinatorial problem is to find the Pareto set of $(f_1(x), f_2(x))$ for $x \in X$.

All multiobjective variants of the shortest path problem, minimum spanning tree problem, assignment problem, knapsack problem and travelling salesman problem are cases of multiobjective combinatorial optimization problems. Mathematical programming and multi–weighted graphs stand out as modelling techniques for these problems (see Section 1.1.2). The reader can refer to Ehrgott & Gandibleux [51] for an extensive survey about multiobjective combinatorial problems.

The coloured–edge graph permits the modelling of multiobjective combinatorial problems by dividing edges in a multi–weighted graph and assigning colours

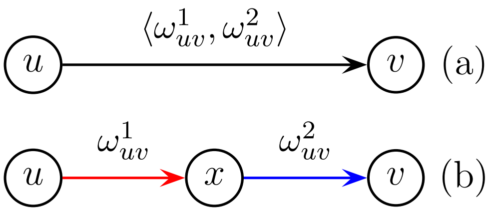


Figure 2.5: Edge division: (a) Initial edge e_{uv}. (b) Division of e_{uv} by adding vertex x.

from the colour set M to each new edge. This is an embedding from the class of all multiweighted graphs into the class of all weighted coloured–edge graphs.

Definition 2.5.1. A multi–weighted graph $\mathbf{G} = \langle V, E, \omega \rangle$ with vertex set V, edge set E and number of weights $r > 1$ (where $r \in \mathbb{Z}$) is a graph such that each edge $e_{uv} \in E$ connecting vertices u and v has r positive weights ω^i_{uv} for $1 \leq i \leq r$, where at least one $\omega^i_{uv} > 0$.

Note that Definition 2.5.1 presumes that at least one of the weight components in each edge is positive, as is common in most logistic applications of multi–weighted graphs. However, an alternative definition of multi–weighted graphs not considered in this thesis drops this requirement and allows negative weight components.

Definition 2.5.2. Edge division of a multi–weighted edge $e_{uv} \in \mathbf{G}$ with r weights is a transformation of e_{uv} by which e_{uv} is split into coloured edges, taking an intermediate edge of colour i and weight ω^i_{uv} for those ω^i_{uv} that are positive. As a result, the multi–weighted edge e_{uv} is replaced by between 1 and r coloured edges.

Definition 2.5.2 has not to be confused with graph subdivision (Gross [52]). Rather, edge division is a mapping for converting a multiweighted graph into a coloured–edge graph. Edge division is exemplified in Figure 2.5 for a biweighted graph edge.

The conversion begins by taking the multi–weighted graph representation $\mathbf{G}$ of the multiobjective combinatorial problem. Next, edge division is applied to $\mathbf{G}$ according to the r weights involved. This is done by splitting $(r - 1)$ times each edge e_{uv}. Next, a label (colour) function $\lambda \colon E \to W$ is applied. Thereby,

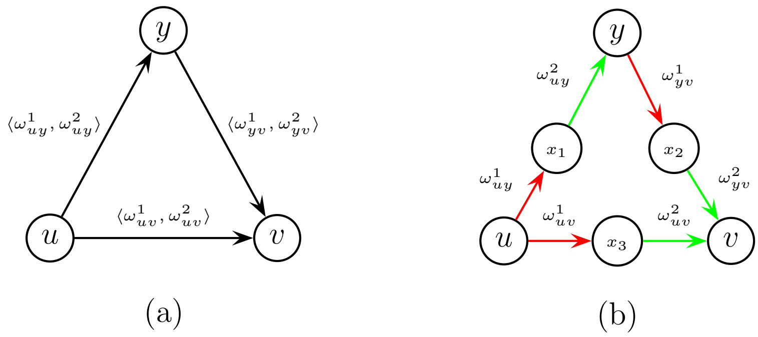


Figure 2.6: Transformation of a biweighted edge graph (a) into a weighted coloured–edge graph (b) by edge division.

each new edge has one specific weight ω_{uv}^i and label i assigned. Notice edge division increases the number of vertices by up to $(r-1)m$, where $m = |E|$. The application of edge division on a multiweighted graph requires not more than rm steps. The use of the coloured–edge graph as a modelling tool for multicriteria combinatorial optimization problems is illustrated in Figure 2.6 with a biweighted graph.

The transformation demonstrates a special fact about multiobjective combinatorial problems. These problems can be found as embeddings in coloured–edge graphs. This makes the coloured–edge graph a nice analytical tool for multiobjective combinatorial problems. For example, upper and lower bounds for a multiobjective combinatorial problem can be established by applying counting arguments on its corresponding coloured–edge graph representation.

2.6 Modelling Multimodal Networks

Literature in multimodal networks reveals that most papers on multimodal networks remove modes from the analysis. In practice, a user of the model is usually unaware of removal of the modes because of the low level of technical knowledge about the model (specially when a mathematical programming approach is employed). An obvious benefit of the coloured–edge graph as a modelling tool is its explicitness and simplicity. A coloured–edge graph is able to deliver a straightforward representation of a multimodal network without incurring a high level of abstraction. All information related to a multimodal system can be easily envisioned by just employing vertices, edges, edge weights and colours.

A decision maker may find a graphical representation useful. Computers do not. A computational representation is needed to make a coloured–edge graph analyzable by computers. For instance, the adjacency matrix and cost matrix can be defined for the coloured–edge graph. A computational representation of the coloured–edge graph implies the study of its computational tractability.

The coloured–edge graphs is seen in this work to typically be tractable without the need to apply any application-specific heuristics or constraints (as commonly occurs in the literature), so can be considered a general tool for the study of multimodal networks. Application-specific considerations can still be applied to the resulting set $\mathcal{M}_{uv}$, or a post-optimal analysis undertaken on it. One facet of this model is that it can be directly applied to multigraph applications, such as transportation networks where there are multiple transportation means between two locations, communication networks where there are multiple links or choice of communication protocols between nodes, or epidemic models which have multiple paths of infection.

However, focusing attention on only the Pareto optimal paths limits the approach to shortest path applications where just the summed contribution of each colour is important, and where any measure of optimality is presumed to be an increasing (linear or non-linear) function of the summed contribution in each colour. For instance, the approach presumes in a transportation network that the optimal path (such as least cost, time, or distance) is some application-specific increasing function of the total weight in each transportation means, or that the user can apply some application-specific criteria to select a preferred path from the Pareto set.

The approach can be adapted for path constraints such as restricting the number of hops or the number of mode changes by slightly enhancing the algorithm used to determine the Pareto set. For instance, besides using colours to represent the different transportation means, an additional colour can be used to count the number of edges in a path as the path is being built during the analysis or to count the number of transfers from one means of transportation to another.

The next chapter investigates the feasibility of the approach as a general tool for multimodal networks by experimentally analyzing the cardinality of the Pareto optimal set $\mathcal{M}_{uv}$. Computations are performed taking several random coloured–edge graphs as input. This allows some conjectures to be made about the tractability of the model. For instance, it is seen later in the thesis that the number of modes k is more of a limiting factor of the approach than is the number of vertices or edges in the graph.

Chapter 3

Computing Pareto Optimal Paths

The number of Pareto optimal paths was presented in previous chapters as a crucial indicator of the computational tractability of the coloured–edge graph. This chapter is devoted to the experimental analysis of this quantity. To carry this analysis out, an algorithm has to be developed to compute optimal paths in a given weighted coloured–edge graph. A generalization of the well–known Dijkstra algorithm will be demonstrated to be adequate for such a task. The algorithm computes the Pareto set for a given weighted coloured–edge graph. Hence, the set $\mathcal{M}_{uv}$ can be studied in order to discover potential patterns in its cardinality. This chapter begins by reviewing literature in algorithmic techniques for the computation of Pareto sets. Next, the multimodal Dijkstra's algorithm is presented together with an experimental study of the cardinality of $\mathcal{M}_{uv}$.

3.1 Introduction

Research in coloured–edge graphs is not extensive. In particular, there is a lack of literature regarding computation of efficient paths in coloured–edge graph. However, some understanding can be extracted by analyzing the embedding of multiobjective optimization problems. For this problem, scientific literature is considerable. Although this problem is not directly addressed by this work, multiobjective combinatorial optimization yields a good starting point for the development of coloured–edge graph algorithms.

Multiobjective combinatorial optimization problems rely on two alternative types of algorithmic techniques for computing efficient solutions. Algorithms are classified as exact when all optimal paths are found, or approximation algorithms when only some solutions representing the entire optimal set or near optimal solutions are found.

Five different approaches are used in the development of exact algorithms. The weighted sum scalarization approach merges objectives into a single weighted function. By varying the weights of the objectives, different efficient solutions can be found. The compromise solution method minimizes the distance from each candidate point to an ideal point. For this approach the Tchebycheff norm is commonly taken as distance measure. The next three approaches are adaptations from single objective optimization. Dynamic programming splits the problem into subproblems which are recursively solved. Algorithms for multiobjective versions of the shortest path problem, knapsack problem and suchlike have been the main "users" of dynamic programming. Particularly regarding algorithms for multiobjective shortest path and knapsack problems, nice overviews are presented by Tarapata [30], Raith & Ehrgott [53] and Martello & Toth [54]. Greedy approaches are also found in the literature. Tung & Chew [55] and Guerriero & Musmanno [56] developed greedy algorithms for determining optimal paths in networks with multiple objectives. Greedy approaches basically take advantage of local estimators for computing optimal solutions of smaller subproblems. Finally, branch and bound is an enumeration technique by which the problem is divided into mutually disjoint and jointly exhaustive subproblems. The efficiency of the method depends strongly on a good estimation of an upper and lower bound for each problem division. The estimation of bounds for branch and bound in multiobjective combinatorial problems is presented by Ehrgott & Gandibleux in [57].

Approximation methods have become very popular in the last three decades. Heuristics and metaheuristics are the main techniques comprising this group. Fast computation and the ability to deal with very hard problems are the mainstays of their popularity. Heuristics are able to determine near–optimal solutions by requiring comparatively few computational resources. Because heuristics are problem–specific, methods developed by this technique cannot solve general instances of multiobjective combinatorial problems. Metaheuristics works by iteratively guiding operations of a subroutine (commonly a heuristic). The subroutine performs a search in the objective space aimed to spot good solutions. The guidance process can be built upon two principles: (1) A subset of the search space is used as initial solution. The search for better solutions is driven by the evolution (adaptation and cooperation) of the population. (2) Starting from an initial solution, a local search is performed on the objective space which is led by a search direction and an aggregation mechanism of the objectives. A complete survey of approximations techniques for multiobjective combinatorial problems is provided by Ehrgott & Gandibleux [58].

Since the coloured–edge graph has a combinatorial structure, techniques previously mentioned for computing optimal solutions (paths) can be considered in the development of algorithms. The next section introduces an algorithm which is capable of identifying optimal paths in a weighted coloured–edge graph by extending the well–know Dijkstra's algorithm.

3.2 Multimodal Network Algorithm

To experimentally study the feasibility of using weighted coloured–edge graphs for multimodal networks an algorithm that determines $\mathcal{M}_{uv}$ is required. A generalization of Dijkstra's algorithm from unimodal networks will be used for this task. In addition, this generalization will be seen to be sufficiently fast to be able to deal with large real multimodal networks.

The classic Dijkstra's algorithm for solving the Single-Source shortest path problem in unimodal networks is a greedy algorithm that uses a priority queue Q to store shortest path estimates from a fixed source vertex s to each vertex v in the network until the shortest path to v is determined. Since the weights of any paths p_{sv} from s to v are linearly ordered there is only at most one shortest path estimate in the queue at a time for each vertex v. At the start of each iteration of the algorithm the shortest path estimate at the front of the queue is the actual shortest path to one of the vertices in the network as all edge weights are presumed to be positive. Dijkstra first published his algorithm in 1959 [59]. Despite its "age", it still enjoys widespread use among algorithm developers because of its efficiency and simplicity. Many extension and variants of this algorithm are found throughout the literature. Nonetheless, generalizations for multimodal networks maintaining mode features throughout its computations have not been found. A recent paper addressing a review and fundamentals about Dijkstra's algorithm is provided by Sniedovich [60].

In a weighted coloured–edge graph Dijkstra's algorithm must be slightly generalized to handle weights of paths being partially ordered rather than linearly ordered. A priority queue Q can again be used to store shortest path estimates with the requirement that if a path p_{sv} from s to v has smaller weight than another path p'_{su} then it should appear earlier in the queue. Although the results presented in this chapter use such a simple queue instead of a more sophisticated non-linear data structure (such as a directed acyclic graph) the performance of the algorithm is seen to be surprisingly good. As in the classical Dijkstra's algorithm the weighted coloured-edge version of the algorithm takes as input a finite graph $\mathbf{G}$ and a source vertex s. It commences at s with the empty path p_{ss} and relaxes each edge that is incident from the source vertex s, adding the single edge paths to the queue. At the front of the queue will be a shortest path estimate p_{sv} to some vertex v adjacent to s. Since all weights are positive in the network p_{sv} must have minimal weight amongst paths from s to v (although it might not be the only minimal path from s to v in the queue), so p_{sv} is added to the set $\mathcal{M}_{uv}$, and removed from the queue. The algorithm then relaxes all the edges incident to v, extending the path p_{sv} by each edge to a path $p'_{su} = p_{sv} \cup \{e_{vu}\}$, adding those extended paths p'_{su} to the queue that have minimal weight amongst paths from s to u, and removing any path p''_{su} from the queue that has greater weight than p'_{su}. The algorithm repeats itself until the queue is empty, producing as output the Pareto minimal set $\mathcal{M}_{sv}$ for each vertex $v \in V$. Hence, the output of the algorithm is in fact $\cup_{v \in V} \mathcal{M}_{sv}$. The following pseudocode describes the algorithm using the notation from [61].

```
Multimodal-Dijkstra(G, s)
▷ Initially no Pareto optimal paths known
for each vertex v
      do M_sv ← NIL
▷ Create a queue Q to hold shortest path estimates during processing
Q ← NIL
add the empty path p_ss from s to s into Q
while Q ≠ ∅
      do remove the path p_sv at front of Q that has some end vertex v
         ▷ Relax the edges incident from v
         for each edge e_vu incident from v
              do ▷ Extend the path p_sv by the edge e_vu
                 p'_su = p_sv ∪ {e_vu}
                 for each p''_su ∈ M_su with greater weight than p'_su
                      do remove the path p''_su from M_su
                 if p'_su has minimal weight in M_su
                    then add p'_su to M_su and to Q
return ∪_{v∈V} M_sv
```

The number of relaxation steps is an important indicator of the algorithm's order, so besides analyzing the cardinality of $\mathcal{M}_{sv}$ for each $v \in V$, the experiments discussed in Section 3.3 also track the number of paths p'_{su} processed by the algorithm.

As an illustration of an application of the weighted coloured–edge graph approach, the algorithm is run with a multimodal network from [62] starting at source vertex 0. Figure 3.1 shows the network which has 21 vertices, 51 edges and 4 different transport choices (bus, metro, private and transfer). The algorithm commences with just the empty path p_{00} on the queue and relaxes two edges: e_{01} with weight $(\text{bus}, \text{metro}, \text{private}, \text{transfer}) = (15, 0, 0, 0)$, and e_{03} with weight $(0, 0, 5, 0)$, which are both added to the queue. Since the two weights are incomparable, either could be at the front of the queue, so the next iteration of the algorithm either adds the path $p_{01} = \{e_{01}\}$ to $\mathcal{M}_{01}$ and relaxes the four incident edges by extending the path p_{01} by each, or else adds the path $p_{03} = \{e_{03}\}$

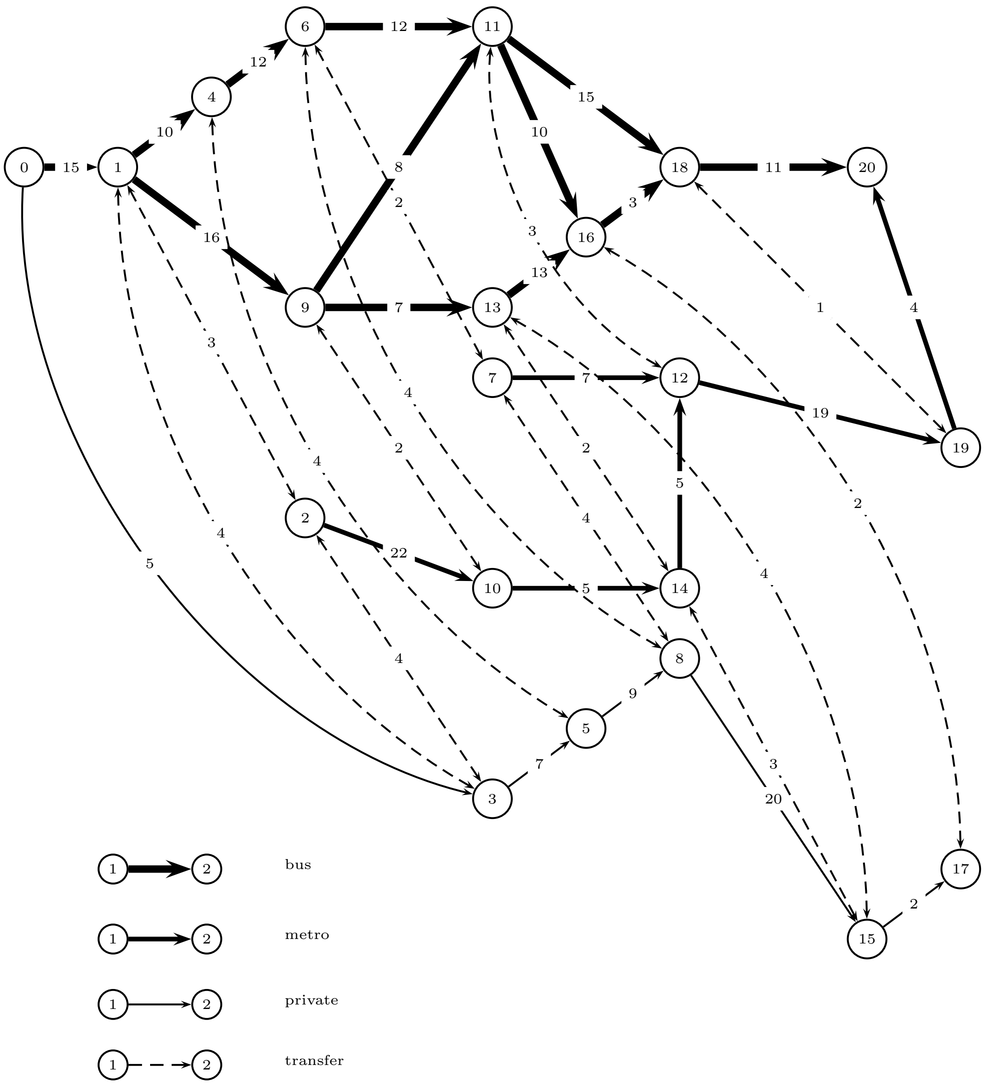


Figure 3.1: Multimodal network from the literature.

to $\mathcal{M}_{03}$ and relaxes the three incident edges by extending the path p_{03} by each. By continuing in this way the Pareto minimal set $\mathcal{M}_{0v}$ is obtained for each vertex v in the network, resulting in 52 Pareto minimal paths from vertex 0 to vertex 20 whose weights are listed in Table 3.1. Depending on the application, constraints or heuristics can then be applied to the 52 paths to select a path preferred by the user. By using just a simple priority queue data structure the generalized Dijkstra's algorithm can determine $\mathcal{M}_{0v}$ for all 21 vertices v within approximately 10ms. The article [62] instead uses a weighted graph approach

Table 3.1: Pareto set for network with 21 vertices and 51 edges.

Path Number	Transport Choice Cost				Cost as per [62]
	Bus	Metro	Private	Transfer	
1	25	4	21	5	55
2	0	30	21	4	55
3	32	9	5	9	55
4	13	11	21	8	53
5	24	0	36	10	70
6	11	26	21	5	63
7	21	26	5	7	59
8	50	19	0	4	73
9	41	4	2	7	54
10	8	45	5	9	67
11	3	4	43	3	53
12	13	4	36	11	64
13	10	30	14	12	66
14	25	26	2	12	65
15	26	0	34	10	70
16	3	31	7	10	51
17	16	4	41	5	66
18	47	9	0	5	61
19	31	31	0	6	68
20	14	0	43	2	59
21	23	45	0	8	76
22	52	4	0	1	57
23	24	7	21	7	59
24	25	11	12	10	58
25	19	9	7	12	47
26	32	22	5	6	65
27	16	31	5	7	59
28	29	27	2	8	66
29	36	0	21	4	61
30	15	4	34	11	64
31	14	27	7	9	57
32	12	23	21	7	63
33	63	0	0	0	63
34	39	23	0	3	65
35	24	23	5	7	59
36	36	26	0	6	68
37	12	30	12	6	60
38	10	26	7	13	56
39	18	31	2	9	60
40	27	0	41	4	72
41	26	4	7	11	48
42	29	0	38	6	73
43	52	0	2	6	60
44	22	30	5	6	63
45	34	9	2	8	53
46	48	0	5	4	57
47	37	4	5	5	51
48	37	30	0	2	69
49	36	7	12	9	64
50	18	4	38	7	67
51	27	27	5	6	65
52	37	0	7	10	54

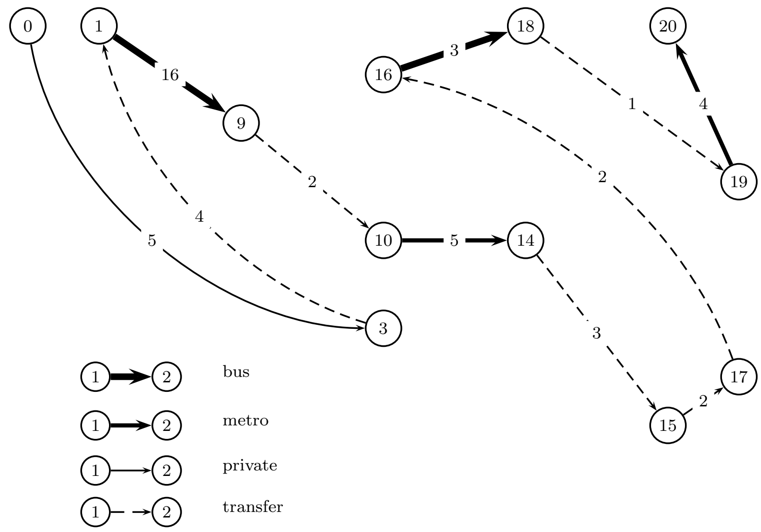


Figure 3.2: Path 25 used for a sensitivity analysis.

with application-specific constraints and a simple cost function which adds the weights in each mode together to get a single–valued total weight, resulting in the paths numbered 2, 25, 33, 47 in the table.

Note that a Pareto set permits a post–optimal analysis to be carried out provided that the total cost is presumed to be an increasing 4–ary function of the summed weight in each mode. For example, suppose in the previous network that the edge weights represent the cost associated to the different means of transport. A natural optimization question could be how much the unit cost associated to a particular mode could be increased or decreased with the current optimal path remaining optimal. As an illustration, path 25 $\{e_{03}, e_{31}, e_{1\,9}, e_{9\,10}, e_{10\,14}, e_{14\,15}, e_{15\,17}, e_{17\,16}, e_{16\,18}, e_{18\,19}, e_{19\,20}\}$ illustrated in Figure 3.2 has least total cost 47, but from the Pareto set it is easily seen that an increase of over 20% in the relative metro costing would make path 41 $\{e_{03}, e_{31}, e_{19}, e_{9\,13}, e_{13\,15}, e_{15\,17}, e_{17\,16}, e_{16\,18}, e_{18\,19}, e_{19\,20}\}$ a better choice, or a 25% increase in bus prices would make path 16 $\{e_{03}, e_{32}, e_{2\,10}, e_{10\,14}, e_{14\,15}, e_{15\,17}, e_{17\,16}, e_{16\,18}, e_{18\,19}, e_{19\,20}\}$ better.

This example demonstrates that the Multimodal Dijkstra's algorithm can quickly calculate the Pareto set without the need to assign relative costs for the different modes. Then alternative cost functions can be evaluated on the paths in the Pareto set or a post–optimal analysis conducted without having to rerun the algorithm.

3.3 Experimental Study

The number of Pareto minimal paths $\mathcal{M}_{uv}$ is investigated for weighted coloured–edge graphs whose weights are randomly chosen according to some input distribution. Moreover, this variable is studied in both a pathological instance of the coloured–edge graph and when arc weights are the Euclidean distance. The information generated about the cardinality of $\mathcal{M}_{uv}$ will help in the establishment of complexity patterns.

3.3.1 Number of processed paths and cardinality of $\mathcal{M}_{uv}$

The objective here is to identify general patterns for the number of processed paths and $\mathcal{M}_{uv}$ cardinality for a vertex $v \in V$. In this test a weighted complete graph $\mathbf{K_n}$ in each colour is taken as input so that each analytical scenario is generated by fixing values for $n = |V|$ and $k = |M|$. Such a graph is characterized by having $kn(n-1)$ edges and the maximum number of possible paths $|\mathcal{P}(u,v)| = \sum_{j=0}^{n-2} \binom{n-2}{j} k^{j+1} j!$ for $v \neq u$, which is a factorial order in n.

Specifically, the algorithm is run for complete graphs with $k = 2, 3, 4, 5$ colours and values of n between 20 and 200. Random edge weights are generated by means of a continuous uniform distribution from the interval $(0, 9]$, although it was seen that the upper bound did not affect the results. The algorithm reports the cardinalities of $\mathcal{M}_{uv}$ for each of the n vertices, but only the cardinality of the final vertex $v = n$ is considered in the analysis.

Figure 3.3 depicts the patterns followed by $\mathcal{M}_{uv}$ cardinality. The graph uses a logarithmic scale for vertical as well as horizontal axes. A logarithmic scale is useful to establish the order of the variables in O–notation. This is done by applying a linear regression analysis between $\log n$ and the logarithm of the studied variable. Thereby, each curve is fitted to a straight line. The slope of this straight line corresponds to the exponent of n. Table 3.2 provides the

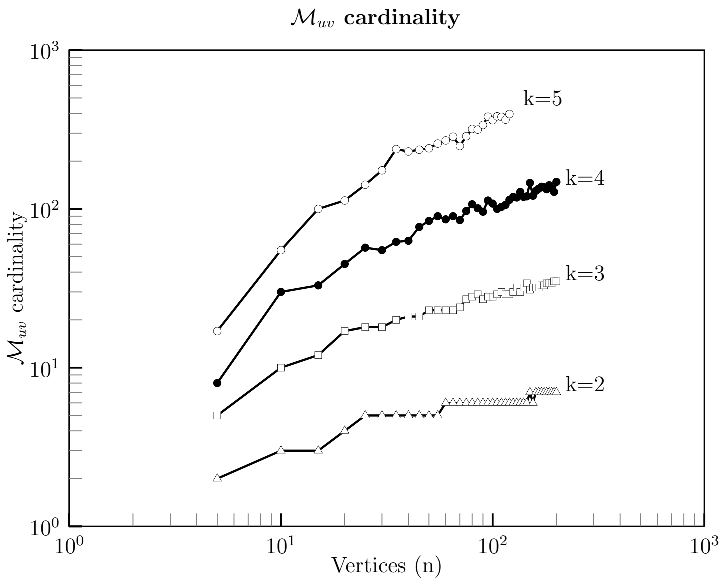


Figure 3.3: Cardinality of $\mathcal{M}_{uv}$ for random weighted coloured–edge graphs with different number of colours k.

Table 3.2: Order of processed paths and $\mathcal{M}_{uv}$ cardinality for several k values.

k	processed paths	$\mathcal{M}_{uv}$ cardinality
2	$O(n^{1.28})$	$O(n^{0.19})$
3	$O(n^{1.37})$	$O(n^{0.32})$
4	$O(n^{1.52})$	$O(n^{0.46})$
5	$O(n^{1.64})$	$O(n^{0.61})$

numerical orders associated to each variable for different k values. These results demonstrate not only that the Pareto optimal set is calculated in polynomial time but also that the resulting set requiring further analysis grows very slowly as a function of n. The results resemble ideas presented by Bentley et al. [63] and Müller–Hannemann et al. [64] for biweighted graphs, suggesting the applicability of the model in real multimodal network scenarios, even when the networks are dense and without having to apply network reduction techniques or heuristics.

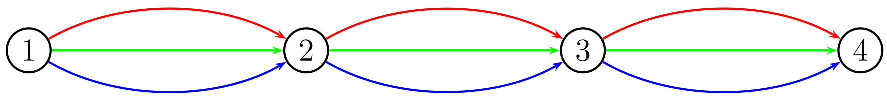


Figure 3.4: Coloured–edge graph with $n = 4$, $s = 2$ and $k = 3$.

3.3.2 Coloured–edge graphs under separation conditions

This experimental group seeks to determine variations in the number of minimal paths when the graph shape is modified by systematically removing edges. To attain this, the separation condition is introduced. This condition forces paths to be comprised of more edges, avoiding the influence that individual edges weights can have on $\mathcal{M}_{uv}$ cardinality. To evaluate the effect of separation on the number of minimal paths, the multimodal Dijkstra's algorithm is applied to graphs that satisfy the separation condition: *There is an enumeration of the vertices $1, 2, \ldots, n$ and a positive integer s called the separation, so that for any edge e_{ij} from vertex i to vertex j one has* $0 < j - i < s$. As an example, Figure 3.4 shows a coloured–edge graph having $n = 4$, $k = 3$ and $s = 2$.

By fixing lower values of s the graph more closely resembles a long chain whose maximum number of edges can be shown to be given by $|E| = k(s-1)(n-s/2)$. This formula can be verified by counting the number of edges E_i from v_i to v_j for i fixed and $j > i$:

$$|E_i| = k \ \min\{s-1, n-i\}.$$

For $i \leq n-s+1$ one has $|E_i| = k(s-1)$ whereas for $i > n-s+1$ $|E_i| = k(n-i)$. Thus,

$$\begin{aligned} |E_1| + \cdots + |E_{n-s+1}| + |E_{n-s+2}| + \cdots + |E_{n-1}| &= k(s-1) + \cdots + k(s-1) \\ &\quad + k(s-2) + \cdots + k \cdot 1 \\ &= k\left((s-1)(n-s+1) + \frac{(s-1)(s-2)}{2}\right) \\ &= k(s-1)(n-s/2). \end{aligned}$$

Separation allows studying the effect that diverse graph shapes have on $\mathcal{M}_{uv}$ cardinality (coloured–edge graphs become thinner as the separation decreases).

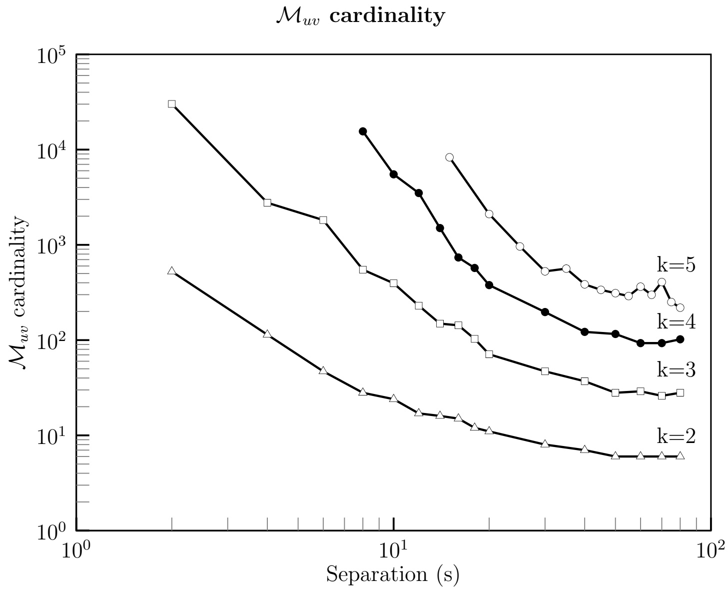


Figure 3.5: Cardinality of $\mathcal{M}_{uv}$ for random weighted coloured–edge graphs with different separation s.

To analyze the effect of the separation condition on the cardinality of the Pareto set, weighted coloured–edge graphs with random weights (uniformly distributed in $(0, 9]$), $|V| = 80$ and k equal to 2, 3, 4 and 5 colours are used to run the Multimodal Dijkstra's algorithm for different separation values. Results are shown in Figure 3.5 which confirms an inverse correlation between separation values and $\mathcal{M}_{uv}$ cardinality. Hence, the more a multimodal network resembles a chain the larger the resulting Pareto set $\mathcal{M}_{uv}$. Unlike other approaches for analyzing multimodal networks, the weighted coloured–edge graph approach often gives smaller Pareto sets on graphs that have greater density.

However, weighted coloured–edge graphs with random weights obeying separation conditions do still exhibit polynomial order. Further experiments with separation values between $s = 2$ and $s = 50$ for weighted coloured–edge graphs with $k = 3$ colours and up to 400 vertices have been undertaken. They indicate that the processed paths have approximate order $O\left(n^{3.5}\right)$ and Pareto set cardinality approximately $O\left(n^{2.5}\right)$, as compared to the earlier results for complete weighted coloured–edge graphs of $O\left(n^{1.37}\right)$ and $O\left(n^{0.32}\right)$ respectively.

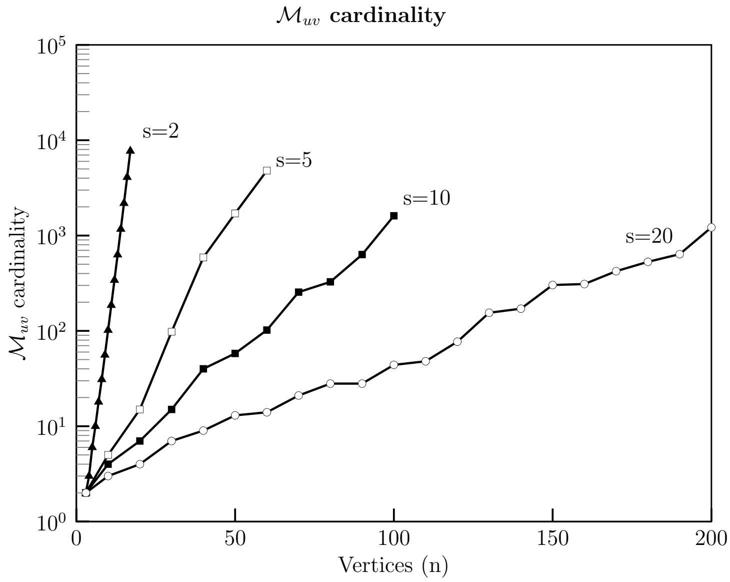


Figure 3.6: Cardinality of $\mathcal{M}_{uv}$ for pathological weighted coloured–edge graphs with $k = 2$ colours and four different separations s.

3.3.3 A pathological instance

In paper [35], Hansen provides an example for the bicriteria shortest path problem that has Pareto set cardinality exponential in n. To demonstrate analogous behaviour is also possible in the weighted coloured–edge graph approach, the graphs from Section 3.3.2 with small separation are modified so that all k edges that have the same initial and terminal vertices are assigned exactly the same random weight (again uniform distributed in $(0, 9]$).

Figure 3.6 shows the pattern followed by $\mathcal{M}_{uv}$ cardinality for separation $s = 2, 5, 10, 20$ and $k = 2$ colours when edges in different colours have the same weights. For this case only the vertical axis has been set as log–scale. This shows that the number of Pareto minimal paths behaves exponentially for such networks, reaching higher orders as k increases. Although this demonstrates the potential exponential behavior of the algorithm and resulting Pareto set, two points should be noted. Firstly, such networks can be considered pathological and rarely would arise in real applications. Secondly, the algorithm can still be feasibly run on moderate size networks with few colours (such as with $|V| = 80$

for $k = 2$ or 3 colours).

3.3.4 Coloured–edge graphs with Euclidean weights

Many real networked systems are conceived only in an abstract "network space" where the exact positions of vertices and edges have no particular meaning. For example, in some cases edges might be an ephemeral representation among vertices, such as business relationships between companies or strategies in a game. However, in many other application fields such as transportation and telecommunications, networks exist in a real three–dimensional space, with vertices having well-defined coordinates. Likewise, edges in these networks are often real physical constructs such as roads or railway lines in transportation networks, optical fibre or other connections in the Internet, cables in a power grid, or oil pipelines. In this context is where the study of networks with Euclidean weights emerges.

Networks with geometric settings can generate large datasets containing thousands of points. For example, a global airline traffic network that considers all airports located around the globe or a national road–rail network whose vertices are road and rail junctions. In such cases, the complexity of the modelling technique is particularly important.

The main experimental input here is a complete coloured–edge graph whose vertices are situated in $\mathbb{R}^b$. Edge weight is set as the Euclidean distance between vertices, which is given by $d(u,v) = \sqrt{\sum_{i=1}^{b}(u_i - v_i)^2}$, where u and v are two vertices located in $\mathbb{R}^b$. A coloured–edge graph that lies in a Euclidean space satisfies three structural properties: (1) The shortest path length in a colour is the Euclidean distance. (2) The triangle inequality is satisfied in each colour. (3) Edges of a minimal pure coloured–edge path do not cross each other.

Two experimental setups are reported in this section. Setup 1 analyzes the cardinality of $\mathcal{M}_{uv}$ for complete coloured–edge graphs whose vertices are randomly generated in $\mathbb{R}$, $\mathbb{R}^2$ and $\mathbb{R}^3$. Setup 2 takes a complete coloured–edge graph in $\mathbb{R}^2$ and perturbs its weights by adding a random variable. Hence, $\omega(e_{uv}) = (1 + \varepsilon r)d(u,v)$, where $\varepsilon = \{0, 0.1, 0.5\}$ and r is a Gaussian random number. Three different values of k are considered (2,3 and 4). The purpose of ε is twofold. First, to emulate a more realistic weight behaviour because distances

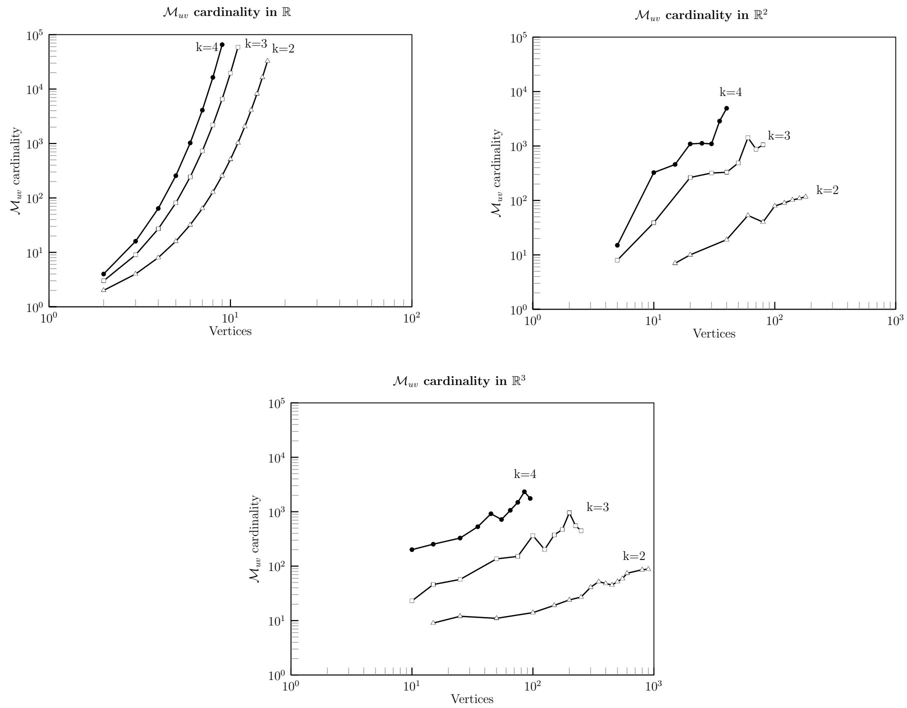


Figure 3.7: $\mathcal{M}_{uv}$ cardinality for coloured–edge graphs in $\mathbb{R}$, $\mathbb{R}^2$ and $\mathbb{R}^3$ with k=2,3 and 4 colours.

in many real transportation systems are not purely Euclidean. Second, to control the amount of departure from a pure Euclidean case.

Each graph in Figure 3.7 describes the variations of $\mathcal{M}_{uv}$ cardinality when generated coloured–edge graphs move from $\mathbb{R}$ to $\mathbb{R}^3$ (Setup 1). A reduction of cardinality is clearly evident as the dimension increases. Likewise, these results suggest the order of $\mathcal{M}_{uv}$ cardinality changes from exponential to polynomial. Although, the nature of the reduction is unexplained at this stage, it certainly yields a promising view regarding applications in 3D environments such as wireless connected networks where colours might represent different wireless protocols.

Figure 3.8 shows results for Setup 2. The cardinality of $\mathcal{M}_{uv}$ decreases as ε increases. In other words, randomness contributes to decreasing the complexity of the model. This is very appealing from an application standpoint because real multimodal networks are predominately subject to some sort of "noise".

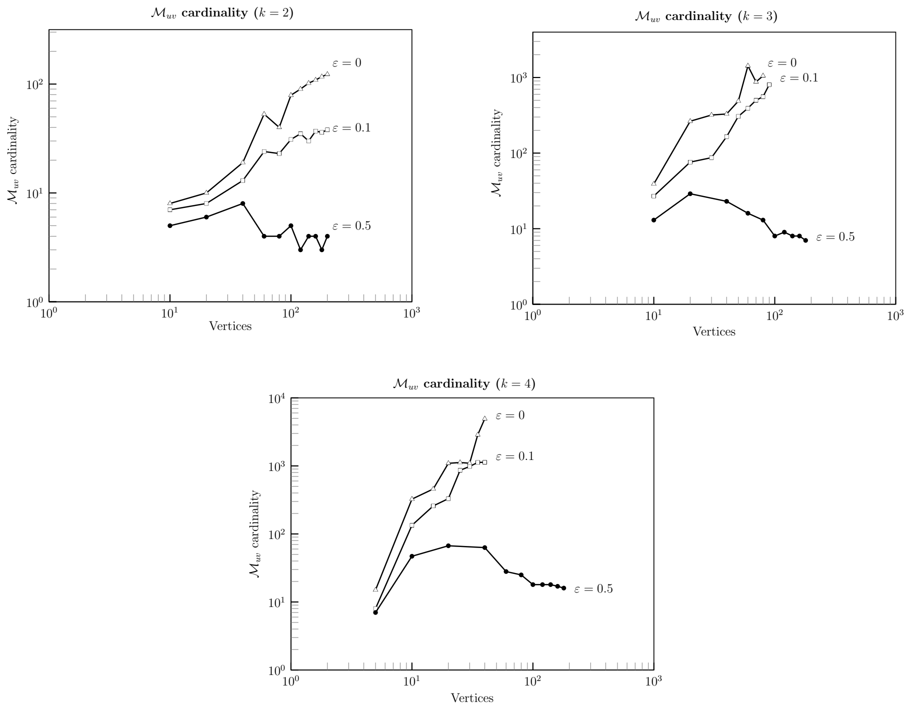


Figure 3.8: $\mathcal{M}_{uv}$ cardinality for coloured–edge graphs with randomly perturbed weights.

3.3.5 Testing some data structures

All the previous experiments were performed using the multimodal Dijkstra's algorithm implemented with a priority queue data structure to hold processed paths. This section is focused on testing alternative data structures for the coloured–edge graph.

The data collected from the experiments in this chapter suggest a link between running time and number of optimal paths. Thus, the ability to efficiently store and process path weights in a colored–edge graph can propel the performance of the approach to higher levels. Two experiments are set for evaluating the performance of two alternative data structures to the queue implementation. Such data structures are called the LIST and the DAG.

The LIST data structure consists of a linked list in which weight tuples are arranged in a linear order. Thereby, when a path is extended (step 12 of the multimodal Dijkstra's algorithm), its associated weight tuple is added to the end of the list to be later processed irrespective of the partial ordering. This list

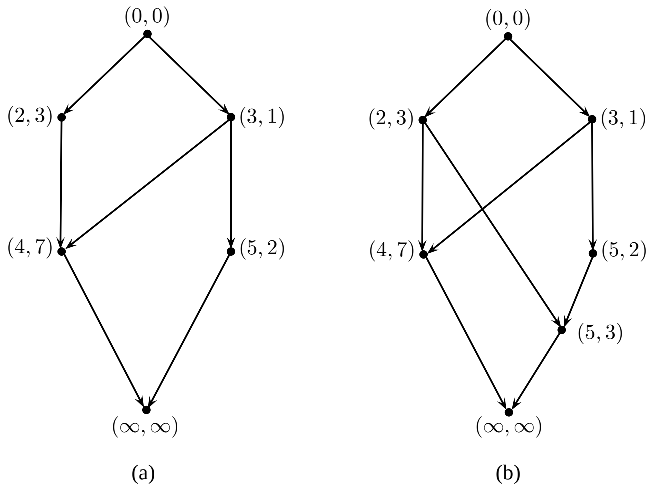


Figure 3.9: Adding tuple $(5, 3)$ to the DAG.

can be implemented as singly or doubly linked. The current analysis employs a doubly linked list.

The DAG is a hierarchical data structure for handling coloured–edge path weights. DAG stands for "Directed Acyclic Graph". Such a graph has its edges arranged according to a partial order relation among weight tuples (DAG vertices). The DAG is a dynamic data structure. Each time a path in a coloured–edge graph is extended (the relaxation step of the multimodal Dijkstra's algorithm) the connectivity is updated (some edges are added whereas others are removed) and a topological order established. This data structure initially has an empty DAG holding only an initial node with tuple $(0, \ldots, 0)$ and a final node with tuple $(\infty, \ldots, \infty)$ for convenience.

Figure 3.9 exemplifies the functioning of the DAG for a coloured–edge graph with two colours.

Assume that at some iteration of the multimodal Dijkstra's algorithm the corresponding DAG is given by graph (a). Next, after applying relaxation the tuple $(5, 3)$ is added so that graph (b) is obtained. The implementation of the DAG scans from $(0, 0)$ forward and from (∞, ∞) backward to identify where a new tuple should be added. Minimal tuples are always the successor of the initial node. These tuples are taken out from the DAG to be inserted in a provisional list.

In case of Figure 3.9, the optimal tuples picked by the DAG implementation would be $(2,3)$ and $(4,0)$. One advantage the DAG has over other data structures is that all the minimal paths can be removed together from the DAG and processed in parallel.

As in previous experiments, the experimental setup here is based on randomly generated coloured–edge graphs. All the graphs are complete. As the number of colours and vertices are the parameters affecting running times, a first test emphasizes the number of vertices over the number of colours, whereas a second oppositely focuses on the number of colours rather than the number of vertices. For the sake of simplicity, such tests are named as TEST 1 and TEST 2.

TEST 1 is based on the number of vertices so that k (number of colours) is fixed to 3. The number of vertices ranges from 10 to 500. Three sets of random weights (each with weights uniformly distributed in $(0,9]$) are generated for each group of vertices. The running time of each data structured is tracked in each set to be subsequently averaged. All this is done by a procedure that first generates the random coloured–edge graph to evoke in turns three subroutines that contain the multimodal Dijkstra's algorithm implemented with the corresponding data structures.

TEST 2 follows the same structure of TEST 1. However, the number of vertices is fixed this time. The chosen number is 20. Although it seems to be a low number of vertices, the advantage is that the number of colours can be made quite large, so the effect of the number of colours can be appreciated more clearly. The number of colours, k, varies from 2 to 11. The procedure described for TEST 1 is also employed here for tracing running times.

Running times for TEST 1 and TEST 2 are shown in Figure 3.10. It can be seen that for TEST 1 the priority queue (QUEUE) as well as the LIST outperform the DAG throughout experiments. The performance of the DAG is comparatively worse as the number of vertices increases. Nonetheless, the experimental orders suggest that running times still are polynomially bounded: $O(n^{2.51})$ for the QUEUE, $O(n^{2.7})$ for the LIST and $O(n^{3.31})$ for the DAG. Meanwhile, TEST 2 reports running times that are almost indistinguishable from each other. Despite the performance of the DAG being worse at the beginning of the experiments,

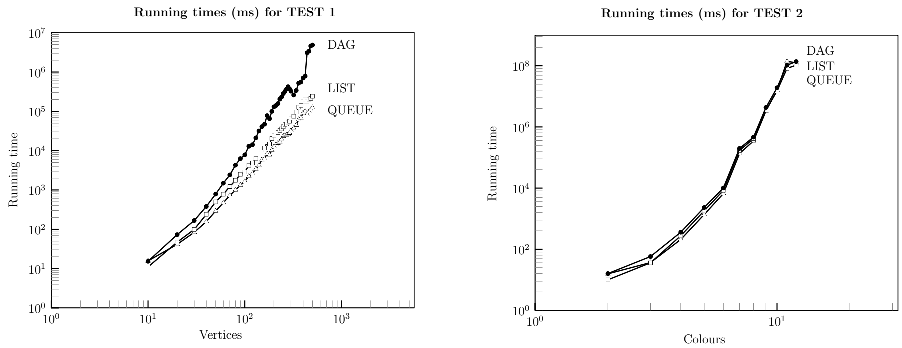


Figure 3.10: Computed running times of TEST 1 and TEST 2.

this improves as the test moves forward. A better understanding of these data structures is obtained by analyzing their asymptotes.

The use of DAG as a data structure to manipulate partially ordered sets has been previously studied by Daskalakis et al. [65]. The hierarchical structure of the DAG is maintained by applying a topological sort on its nodes. This make any new insertion highly costly. For the DAG an insertion is $O(k\varrho v)$ where ϱ is the average width of the DAG and v is the number of levels. Since roughly v is somewhere between $\log \varrho q$ and p/ϱ (q is the average number of minimal paths), the final running time should be between $O(k\varrho \log \varrho q)$ and $O(kq)$.

In case of the LIST, the running time is determined by the time spent in keeping the data structure free of domination. Each new tuple has to be compared to those already in the processed set. Such operation is performed in $O(kp)$ time.

The priority queue was implemented as a heap. If all paths were comparable, the order of the running time for this data structure would be $O(k \log p)$. Recall that a totally ordered priority queue just requires $O(\log n)$ time for any operation, where n is the number of nodes in the heap.

The main conclusion obtained from random instances of the coloured–edge graphs is that QUEUE provides better performance over the LIST and DAG when vertices are the pervasive parameter. The QUEUE remains as a good option when the number of colours is more important. Nonetheless, the DAG and LIST are capable of comparable running times as the number of colours increases.

It still needs to be analyzed whether the behaviour of these three data structures remains similar in real multimodal networks. This question is tackled in Chapter 6.

3.4 Remarks about The Experimental Study

The main purpose of the experimental study was to identify complexity patterns in the number of optimal paths ($\mathcal{M}_{uv}$ cardinality).

The number of optimal paths observed in randomly weighted coloured–edge graphs are significantly smaller than those encountered in pathological instances. Separation conditions also seem to support such results. As the value of s increases, the number optimal paths seems to go down. This leads to questions regarding the influence that "chainlike" coloured–edge graphs have on the number of optimal paths. It can be noticed that if the density of a coloured–edge graph increases, in general its number of minimal paths decreases. Nevertheless, there is another factor that needs to be taken into account. Weight patterns also affect the cardinality of $\mathcal{M}_{uv}$. For some specific weight configurations, the variable $\mathcal{M}_{uv}$ might go up or down. Experiments in pathological graphs apparently confirm such ideas as well.

Tests in Euclidean spaces provided understanding about the connection between graph shape and number of optimal paths. More optimal paths were obtained for coloured–edge graphs in $\mathbb{R}$. A graph in one dimensional space resembles a chain. As the tests increased the number of dimensions, the order of the cardinality decreased. This can be interpreted as re-shaping of a coloured–edge graph to a more branched configuration. Similarly, a coloured–edge graph subject to separation gets re–shaped because a thinner graph is obtained when the values of separation decrease. The action of randomness on weights unveiled the existence of a potential mechanism that reduces the cardinality of $\mathcal{M}_{uv}$. For instance, a pathological instance can generate a lower number of optimal paths by adding random variables to its weights.

Orders are supporting the previous ideas throughout the experiments. An increment is noticed when separation decreases reaching exponential orders for some instances. Likewise, pathological chains provided a high number of optimal paths, producing orders that might be considered exponential. Finally, experiments in Euclidean spaces also reported exponential orders for coloured–edge graphs in one dimension. In the same way, orders in the cardinality of $\mathcal{M}_{uv}$ were lower when the amount of randomness added to weights was increased.

In consequence, the experimental study allows some ideas about the cardinality of the $\mathcal{M}_{uv}$ to be conjectured:

1. The maximum cardinality of $\mathcal{M}_{uv}$ is at least exponential in n.
2. The average cardinality of $\mathcal{M}_{uv}$ is polynomially bounded.
3. An exponential order of $\mathcal{M}_{uv}$ rarely occurs in real instances. In other words, real implementations of the coloured–edge graph are almost always tractable.

Subsequent chapters address these conjectures as well as other tractability issues of the colured–edge graph.

Chapter 4

Upper Bounds on The Number of Minimal Paths

This chapter addresses the computation of a tight upper bound on the cardinality of a minimal set of paths. Three theorems are introduced with their corresponding proofs. The bounds obtained indicate that weighted coloured–edge graphs are able to generate at most an exponential number of minimal paths. An additional problem studied in this chapter is related to the result established by Hansen [35] for the bicriteria shortest path problem. Basically, Hansen proved the existence of a biweighted graph with an exponential number of optimal paths. However, two constructions presented here might question how close Hansen's construction is to a worst case.

4.1 Introduction

Definition 2.2.1 states that a weighted coloured-edge graph $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ is composed of a directed multigraph $\langle V, E \rangle$ with vertex set V and edge set E, a weight function $\omega \colon E \to \mathbb{R}^+$, and a colour function $\lambda \colon E \to M$, where M is a set of colours for the edges. Associated with each edge $e \in E$, there is an initial vertex $u \in V$ and a terminal vertex $v \in V$, a positive weight $\omega(e) \in \mathbb{R}^+$, and a colour $\lambda(e) \in M$.

Definition 2.3.3 indicates that a path in $\mathbf{G}$ consists of a finite set of edges $\{e_i \mid 1 \leq i \leq l\}$ for which the initial vertex of each e_{i+1} is the terminal vertex of

e_i. The path is called *simple* if no two edges in the path have the same initial vertex nor the same terminal vertex. It is straightforward to verify that a finite coloured–edge graph has at most the following number of simple paths from a chosen vertex u to a vertex v:

$$k + k^2(n-2) + k^3(n-2)(n-3) + \cdots + k^{n-1}(n-2)(n-3)\cdots 1.$$

For any path p_{uv} from a vertex u to a vertex v and any colour $c \in M$ the path weight in colour c is defined by $\omega_c(p_{uv}) = \sum_{e_{xy}\in p_{uv}, \lambda(e_{xy})=c} \omega(e_{xy})$, namely the sum of the weights for all the edges that have colour c. The *weight* of a path is represented as a k-tuple $(\omega_{c_1}(p_{uv}), \omega_{c_2}(p_{uv}), \ldots, \omega_{c_k}(p_{uv}))$, giving the total weight of the path in each colour. A preorder $\leq$ can be defined on the paths from u to v by $p_{uv} \leq p'_{uv}$ if for every colour c one has $\omega_c(p_{uv}) \leq \omega_c(p'_{uv})$, essentially using the partial order defined on the weights by the product partial ordering on $\mathbb{R}^k$.

This chapter is interested in establishing bounds on the number of paths that can be minimal. Specifically, for a finite weighted coloured-edge graph $\mathbf{G}$ with vertices u and v, what is the number of path weights that can be minimal. Certainly, if the graph is disconnected then there might be no paths from u to v, and if it is connected with edges in each allowed colour then there are at least k minimal paths, since paths that have edges solely in a single colour are incomparable. However, an upper bound might appear to be difficult to establish. The interest in this problem arises from multimodal network applications, which can be modeled using weighted coloured-edge graphs, where the weights represent some form of cost in a network (such as distance) and the colours represent the mode of transportation. In most multimodal optimization applications constraints need to be placed on the network to make the determination of a shortest path fulfilling application-specific criteria tractable. Often such criteria are based on the path weights. So if the set of minimal paths has manageable cardinality then the application-specific criteria might just be applied to the set of minimal paths.

As a simple example, consider the graph in Figure 4.1 with four vertices, fourteen edges and two possible colours. There are 26 paths to consider from u to v. After carefully checking through these 26 paths it can be seen that there are between two and a maximum of eight minimal path weights possible, depending

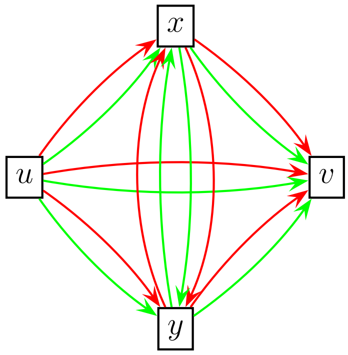


Figure 4.1: Coloured–edge graph with 4 vertices, 14 edges and 2 colours.

on the weights of the edges in the graph. The mechanics in the computation of minimal paths yields an argument to establish a tight lower bound on the number of minimal paths for complete coloured–edge graphs.

Proposition 4.1.1. *For a finite coloured-edge graph* $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ *connected by a path from a source vertex* u *to a destination vertex* v *in each colour, the number of minimal paths in* $\mathcal{M}_{uv}$ *is at least* k*.*

Proof. Assume that $\mathbf{G}$ is connected in each colour from u to v. Then, there exists a path purely in each colour from u to v so there must be a minimal path purely in that colour. Therefore, the graph $\mathbf{G}$ must have at least k minimal paths from u to v. □

An example of Preposition 4.1.1 is Figure 2.4 in Chapter 2 where two minimal pure coloured–edge paths in colours red and green are found.

As an important special case of weighted coloured-edge graphs a graph is called a *chain* if in some enumeration of its vertices $v_1, v_2, v_3, \ldots, v_{n-1}, v_n$ the graph only has edges from a vertex v_x to the next vertex v_{x+1} in the enumeration. Figure 4.2 illustrates a chain with n vertices and $k = 3$ colours.

Consider the construction of Figure 4.3 for k colours (for sake of simplicity, just three colours are shown). This is called an *exponentially weighted coloured–edge chain.* This chain has k^{n-1} minimal paths from v_1 to v_n as the following proposition claims.

Proposition 4.1.2. *For an exponentially weighted coloured–edge chain with* n *vertices and* k *colours, the number of minimal paths is* k^{n-1}*.*

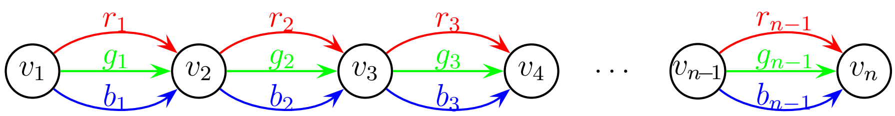


Figure 4.2: Coloured–edge chain with n vertices and $k = 3$ colours.

Proof. The proof goes by induction on n. Let $h_k(n)$ be the number of minimal paths in an exponential coloured–edge chain with n vertices ($n \geq 1$) and fixed number of colours k. The base case $h_k(1) = 1$ clearly holds. For inductive hypothesis, assume that $h_k(m) = k^{m-1}$ holds for any exponential weighted coloured–edge chain with m vertices. Consider now an exponentially weighted–coloured–edge chain with $m+1$ vertices and consider two distinct paths $p = \{e_{12}, e_{23}, \dots, e_{m\,m+1}\}$ and $p' = \{e'_{12}, e'_{23}, \dots, e'_{m\,m+1}\}$ from v_1 to v_{m+1}. To show that p and p' are incomparable, consider two possible cases: (i) If $e_{m\,m+1} = e'_{m\,m+1}$ so p and p' share the same edge from v_m to v_{m+1} then $p = \{e_{12}, e_{23}, \dots, e_{m-1\,m}\}$ and $p' = \{e'_{12}, e'_{23}, \dots, e'_{m-1\,m}\}$ are distinct paths from v_1 to v_m so by inductive hypothesis they must be incomparable. Hence p and p' are incomparable too. (ii) If $e_{m\,m+1} \neq e'_{m\,m+1}$ so the edges $e_{m\,m+1}$ and $e'_{m\,m+1}$ from v_m to v_{m+1} with weight 2^{m-1} have different colours c and c' then $\omega_c(p) \geq 2^{m-1} > 1+2+4+\ldots+2^{m-2} \geq \omega_c(p')$. Whereas $\omega_{c'}(p) \leq 1+2+4+\ldots+2^{m-2} < 2^{m-1} \leq \omega_c(p')$. Hence p and p' are incomparable and $h_k(m+1) = h_k(m) \times k = k^{(m+1)-1}$. $\square$

Adding more *forward* edges to the chain from x to y for $y > x+1$ increases the number of paths in the graph but it cannot increase the number in a minimal set of incomparable paths, and might possibly decrease the number. But adding *backward* edges from x to y for $y < x$ appears to greatly complicate the situation. However, it is shown in Section 4.3 that essentially chains illustrate the worst possible situation, whereas more general weighted coloured-edge graphs may have

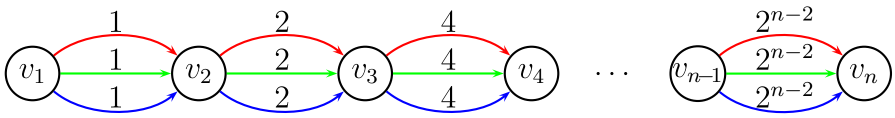


Figure 4.3: An exponentially weighted coloured–edge chain.

a factorial number of paths a minimal set of incomparable paths can only have cardinality up to k^{n-1}. Hence in a multimodal network this is a tight upper bound on the Pareto set of minimal path weights that might need to be considered when applying application-specific constraints.

The number of minimal paths can also be estimated for coloured–edge chains whose path weights are given by the number of edges. To show this, two lemmas need to be firstly introduced.

Lemma 4.1.3. *Let $n \geq 0$ and $k \geq 1$. Then the following combinatorial equality holds*

$$\binom{n+k-1}{k-1} + \binom{n+k-2}{k-1} + \cdots + \binom{k-1}{k-1} = \binom{n+k}{k}. \tag{4.1}$$

Proof. Use induction on n for fixed k. For $n = 0$ the equality holds since $\binom{k-1}{k-1} = 1 = \binom{k}{k}$. Suppose for some $m \geq 0$ that $\binom{m+k-1}{k-1} + \binom{m+k-2}{k-1} + \cdots + \binom{k-1}{k-1} = \binom{m+k}{k}$ holds. Then for $m+1$

$$\begin{aligned}\binom{m+1+k-1}{k-1} + \binom{m+k-1}{k-1} + \binom{m+k-2}{k-1} + \cdots + \binom{k-1}{k-1} &= \binom{m+k}{k-1} + \binom{m+k}{k} \\ &= \frac{(m+k)!}{k!(m+1)!} \cdot (k+m+1) \\ &= \binom{m+1+k}{k}.\end{aligned}$$

Hence equality 4.1 holds for $m+1$. ☐

Lemma 4.1.4. *Let*

$$S_{nk} = \{(x_1, x_2, \ldots, x_k) \mid x_i \in \mathbb{Z},\ x_i \geq 0,\ \forall i = 1, \ldots, k \text{ and } \sum_{i=1}^{k} x_i = n\}.$$

Then

$$|S_{nk}| = \binom{n+k-1}{k-1} = \frac{(n+k-1)!}{(k-1)!n!}. \tag{4.2}$$

Proof. Induction on k is used to prove that

$$|S_{nk}| = \binom{n+k-1}{k-1}\ \forall n \geq 0.$$

For $k = 1$ and any n, $S_{nk} = \{(x_1) \mid x_1 \in \mathbb{Z},\ x_1 \geq 0 \text{ and } x_1 = n\}$ so $|S_{nk}| = 1$. For the induction hypothesis, assume for some j that $|S_{nj}| = \binom{n+j-1}{j-1}$ $\forall n \geq 0$

and for any $n \geq 0$ consider the set $S_{n\,j+1} = \{(x_1, x_2, \ldots, x_{j+1}) \mid x_i \in \mathbb{Z},\ x_i \geq 0,\ \forall i = 1, \ldots, k$ and $\sum_{i=1}^{j+1} x_i = n\}$. For any tuple $(x_1, x_2, \ldots, x_{j+1}) \in S_{n\,j+1}$ one has that $\sum_{i=1}^{j} x_i + x_{j+1} = n$ so $x_{j+1} \leq n$. Thus $(x_1, x_2, \ldots, x_j) \in S_{n-x_{j+1}\,j}$, for which there are $\binom{n-x_{j+1}+j-1}{j-1}$ such tuples. So by counting over all the possible values $0, 1, 2, \ldots, n$ for x_{j+1}, there are

$$|S_{n\,j+1}| = \binom{n-0+j-1}{j-1} + \binom{n-1+j-1}{j-1} + \binom{n-2+j-1}{j-1} + \cdots + \binom{n-n+j-1}{j-1}$$

distinct tuples in $S_{n\,j+1}$. Hence $S_{n\,j+1} = \binom{n+j}{j}$ by Lemma 4.1.3. Hence equality 4.2 is true by induction.
Alternatively, this result can instead be proved by considering the computation of S_{nk} as a sampling of k items n times with replacement without ordering which is a well–known combinatorics problem. $\square$

As a corollary of the last lemma the number of minimal paths for a k coloured–edge chain with n vertices is obtained.

Corollary 4.1.5. *For a coloured–edge chain with n vertices and k colours where each edge has weight 1, a maximal set of incomparable minimal paths has cardinality $f(n,k)$ given by*

$$f(n,k) = \binom{n+k-2}{k-1}. \tag{4.3}$$

Proof. It is enough to show that any tuple $(x_1, x_2, \ldots, x_k)$ where $\sum_{i=1}^{n} x_i = n-1$ is attainable by some path p in the chain and all such tuples are incomparable. The path p can be constructed to have weight $(x_1, x_2, \ldots, x_k)$ by taking the first x_1 edges of the path in colour c_1, then the next x_2 edges in colour c_2 and so on. Furthermore, any two distinct tuples $(x_1, x_2, \ldots, x_k)$ and $(x'_1, x'_2, \ldots, x'_k)$ must be incomparable as $\sum_{i=1}^{k} x_i = n-1 = \sum_{i=1}^{k} x'_i$. $\square$

Note that if for all colours the edges in a particular colour have the same weight, then equality 4.3 still holds for a k–coloured chain with n vertices.

The pattern in $f(n,k)$ is identified by tabulating this function for small values of k and taking an arbitrary fixed n (see Table 4.1). Observe that the number of minimal paths for a fixed k is given by $n^{\overline{k-1}}/(k-1)!$ for $k > 1$. The term $n^{\overline{k-1}}$ corresponds to a raising factorial power of n (see Graham et al. [66]).

Table 4.1: Values of $f(n,k)$ for several values of k.

k	$f(n,k)$
2	n
3	$\frac{n(n+1)}{2}$
4	$\frac{n(n+1)(n+2)}{6}$
5	$\frac{n(n+1)(n+2)(n+3)}{24}$

4.2 Canonical Graphs

To prove that k^{n-1} is an upper bound on the cardinality of the Pareto set of minimal paths a special class of weighted coloured-edge graph is first introduced.

Definition 4.2.1. A weighted coloured-edge graph $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ is called *canonical* if:

- $\mathbf{G}$ is *complete* in each colour, namely for all vertices $x \neq y$ and colour c there is exactly one edge e_{xy} from x to y with $\lambda(e_{xy}) = c$,
- $\mathbf{G}$ satisfies the *triangle inequality* in each colour, namely for all distinct vertices x, y, z and colour c, the triangle formed by the three edges e_{xy}, e_{yz}, e_{xz} with $\lambda(e_{xy}) = \lambda(e_{yz}) = \lambda(e_{xz}) = c$ obeys $\omega(e_{xz}) \leq \omega(e_{xy}) + \omega(e_{yz})$.

The following lemma shows that it will be sufficient to establish the bound on the class of canonical graphs.

Lemma 4.2.1. *Given any finite weighted coloured-edge graph $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ there is a canonical graph $\mathbf{G}^* = \langle V, E^*, \omega^*, \lambda^* \rangle$ with the same vertices and colours as $\mathbf{G}$, where $E \subseteq E^*$, $\lambda^*|_E = \lambda$, and for which every minimal path in $\mathbf{G}$ is also minimal in $\mathbf{G}^*$.*

Proof. Firstly note that $\mathbf{G}$ can be made complete by adding edges e_{xy} with weight $n \cdot w$ where $n = |V|$ and w is the maximum weight of any edge in $\mathbf{G}$. The added edges won't affect any existing minimal paths as they contain at most $n-1$ edges so their path weight in each colour is less than $n \cdot w$. It might however introduce additional minimal paths in the graph if there were no existing paths from x to y in some colour. Take E^* to be the resulting set of edges. Next, the graph can have its weights altered by defining $\omega^*(e_{xy})$ to be the weight of the shortest path from x to y that only uses edges with colour $\lambda^*(e_{xy})$. Thanks to completeness ω^* is well-defined and clearly the resulting graph satisfies the triangle inequality. $\square$

4.3 Upper Bounds on Minimal Paths

Theorem 4.3.1. *Suppose* $\mathbf{G}$ *is a weighted coloured-edge graph with* $n \geq 2$ *vertices and* k *colours. Then a set of incomparable minimal paths in* $\mathbf{G}$ *from one vertex to another can have at most* k^{n-1} *paths.*

Proof. The proof uses a counting argument and induction to bound the cardinality $f_c(n)$ of a set of incomparable minimal paths whose first edge has colour c in any weighted coloured-edge graph with n vertices. Trivially, $f_c(2) = 1$ for any graph $\mathbf{G}$ with only two vertices.
For the inductive step presume that $f_c(m) \leq k^{m-2}$ in any graph with $m \leq n$ vertices and suppose $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ has $n+1$ vertices. By Lemma 4.2.1 $\mathbf{G}$ can be presumed to be canonical. Let u and v be any two distinct vertices of $\mathbf{G}$ and S be a set of incomparable minimal paths from u to v. By the triangle inequality it can be presumed that no minimal path in S has two consecutive edges of the same colour. Furthermore, a useful observation for a minimal path that starts with an edge e_{ux} of colour $\lambda(e_{ux}) = c$ and which passes through some vertex y before reaching v is that for the edge e_{uy} with $\lambda(e_{uy}) = c$ minimality of the path ensures that $\omega(e_{ux}) < \omega(e_{uy})$. To show that $f_c(n+1) \leq k^{n-1}$ for each colour c order the remaining vertices of V, $v_1, v_2, \ldots, v_{n-1}$ so that if $i < j$ then $\omega(e_{uv_i}) \leq \omega\left(e_{uv_j}\right)$ where e_{uv_i} and e_{uv_j} are the edges of colour c from u to v_i and v_j respectively. By the earlier observation, no minimal path that starts with the edge e_{uv_i} of colour c can pass through any of the vertices v_j for $j < i$. Hence, any minimal path that starts with the edge $e_{uv_{n-1}}$ has only a choice of $k-1$ edges to reach v (since its consecutive edges are not of the same colour), so there are at most $k-1$ such paths. Similarly, any minimal path that starts with the edge e_{uv_i} can only utilize $v_i, v_{i+1}, \ldots, v_{n-1}$ and v, so by the inductive hypothesis for $m = n-i+1$ there are at most $\Sigma_{c' \neq c} f_{c'}(n-i+1) \leq (k-1)k^{n-i-1}$ paths. Summing across all the edges $e_{uv_1}, e_{uv_2}, \ldots, e_{uv_{n-1}}$ and e_{uv} gives $f_c(n+1) \leq (k-1)k^{n-2} + (k-1)k^{n-3} + \cdots + (k-1) + 1 = k^{n-1}$. Since there are k possible colours in which to start a path this completes the proof. □

Note that as the proof relies on being able to linearly order the vertices $v_1, v_2, \ldots, v_{n-1}$ based on the edge weights $\omega(e_{uv_i})$ in a specific colour c, the proof can not be readily adapted to arbitrary multiweighted graphs. This result gives a tight upper bound on the cardinality of a set of incomparable minimal paths in a weighted coloured-edge graph. This bound is suitable for applications that are primarily interested in determining a minimal path given criteria that depend on the total weight in each mode of transportation. However, the proof can be slightly modified to provide a bound on the total number of minimal paths in the graph from u to v, counting all minimal paths that are comparable with each other (having the same path weight).

Theorem 4.3.2. *Suppose* $\mathbf{G}$ *is a weighted coloured-edge graph with* $n \geq 2$ *vertices and* k *colours for which there is only at most one edge of each colour between vertices. Then* $\mathbf{G}$ *has at most* $k(k+1)^{n-2}$ *minimal paths in* $\mathcal{M}_{uv}$ *from a source vertex* u *to a destination vertex* v.

Proof. The proof is similar to that of Theorem 4.3.1 except that the counting argument bounds $g_c(m) \leq (k+1)^{m-2}$ and paths are allowed to have the same colour on two consecutive edges. Thus, the inductive step takes $g_c(m) \leq (k+1)^{m-2}$ in any graph with $m \leq n$ vertices. Suppose $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ has $n+1$ vertices and it is canonical. Let u and v be any two vertices of $\mathbf{G}$ and S be a set of incomparable minimal paths from u to v. To show that $g_c(n+1) \leq (k+1)^{n-1}$ for each colour c vertices $v_1, v_2, \ldots, v_{n-1}$ are sorted so that $\omega\left(e_{uv_i}\right) \leq \omega\left(e_{uv_j}\right)$ for $i < j$. Here, e_{uv_i} and e_{uv_j} are the edges of colour c from u to v_i and v_j respectively. As in the previous theorem, no minimal path that starts with the edge e_{uv_i} of colour c can pass through any of the vertices v_j for $j < i$. Therefore, any minimal path that starts with the edge $e_{uv_{n-1}}$ has only a choice of k edges to reach v, so there are at most k such paths. Analogously, any minimal path that starts with the edge e_{uv_i} can only utilize $v_i, v_{i+1}, \ldots, v_{n-1}$ and v, so by the inductive hypothesis for $m = n-i+1$ there are at most $\Sigma_{c' \neq c} g_{c'}(n-i+1) \leq k(k+1)^{n-i-1}$. Adding up across all the edges $e_{uv_1}, e_{uv_2}, \ldots, e_{uv_{n-1}}$ and e_{uv} gives,

$$\begin{aligned} g_c(n+1) &\leq k(k+1)^{n-2} + k(k+1)^{n-3} + \cdots + k(k+1) + k + 1 \\ &\leq k \cdot \frac{(k+1)^{n-1} - 1}{(k+1) - 1} + 1 \\ &\leq (k+1)^{n-1}. \end{aligned}$$

The desired bound is obtained by considering k possible colours. $\square$

The bound established in Theorem 4.3.2 can be seen to be tight by constructing examples based on the chain example in Section 4.1 but with additional edges, such as the example for $n = 4$ shown by Figure 4.4 in which each of the 3×4^2 paths from v_1 to v_4 is minimal.

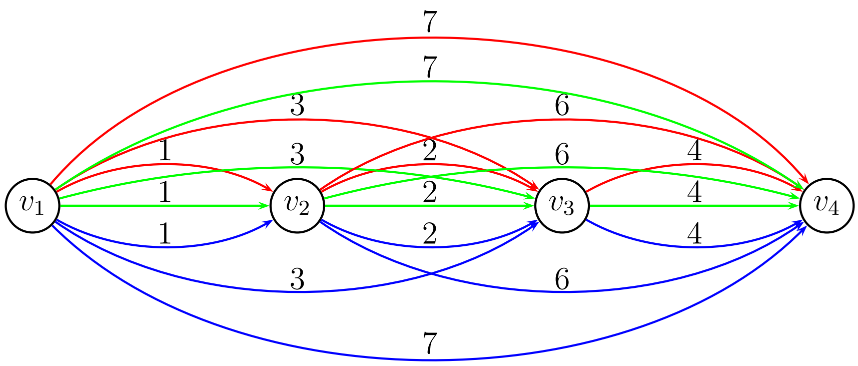


Figure 4.4: Example of a weighted coloured–edge graph in which each path from v_1 to v_4 is minimal.

The calculated bounds provide a pessimistic view about the tractability of the coloured–edge graph as a modelling technique. To appreciate the difficulty of computing minimal paths in a coloured–edge graph, the NP–completeness of the problem is next studied. Recall first that the well–known Bin Packing Problem is $\mathcal{NP}$–complete:

Bin packing problem

Instance: A set N of n items, each with a positive integer weight ω_i for $1 \leq i \leq n$, a positive number of bins k and a positive integer bin capacity β_j, for $1 \leq j \leq k$.
Question: Can the set N be partitioned in k subsets such that for each subset, the sum of the weights ω_i in partition i is at most β_j?

By using a reduction from the bin packing problem, the determination of minimal paths in a coloured–edge graph can be shown to be $\mathcal{NP}$–complete.

Restricted minimal paths in a coloured–edge graph

Instance: a coloured–edge graph $\mathbf{G} = \langle V, E, \omega, \lambda \rangle$ with k colours, two distinguished vertices s and t and a maximum path weight $\alpha = (\alpha_1, \ldots, \alpha_k)$.
Question: Does there exist a path from s to t with total weight $\leq \alpha$?

Theorem 4.3.3. *The restricted minimal path problem in a coloured–edge graph is $\mathcal{NP}$–complete.*

Proof. Use a reduction from the Bin Packing Problem. Consider a coloured–edge chain with k colours, $n+1$ vertices, $s = v_1$ and $t = v_{n+1}$. Assign each item weight ω_i to each edge from v_i to v_{i+1}. Note that each path joining s and t has a weight tuple indicating the total weight assigned to each bin. Hence, each path from s to t in a coloured–edge chain is a partition that might solve the bin packing problem. Next, set $\alpha = (\beta_1, \beta_2, \ldots, \beta_k)$ and apply the condition that the weight of a path from s to t in the chain has not to be greater than α. $\square$

4.4 Hansen's Result for Biweighted Graphs

As far as the literature about multicriteria combinatorial optimization problems indicates, the multicriteria version of the shortest path problem is intractable even for two criteria. This assertion was originally proved by Hansen [35]. In his paper a family of graphs were given for which the number of efficient paths in a minimal complete set grows exponentially with the number of vertices. As a complement, Serafini [67] later demonstrated the $\mathcal{NP}$–completeness of this problem. He did this by constructing a reduction from the 0–1 Knapsack problem to the Single–Source Shortest Path Problem. No further analysis has been undertaken until now regarding these results.

This section aims to re–analyze the result introduced by Hansen and compares it with the coloured–edge graph bound. As shown in Chapter 2, the coloured–edge graph can be utilized for representing multicriteria combinatorial optimization problems. Due to this, this graph is an alternative tool for the determination of bounds on multicriteria optimization problems.

To begin, the theorem developed by Hansen is presented.

Theorem 4.4.1 (Hansen [35])**.** *The bicriteria shortest path problem is, in worst case, intractable, i.e. require for some problems a number of operations which grows at least exponentially with these problem's characteristics.*

Proof. It is sufficient to show there exists for each problem a family of graphs for which the number of efficient paths in the Pareto set grows exponentially with n. Consider a family of graphs $\mathbf{G}$ with an odd number n of vertices and $3(n-1)/2$ edges defined as follows: each vertex v_i, with $i = 1, 3, \ldots, n-2$ is the start of an edge $e_{v_i\, v_{i+2}}$ with weight $(2^{\frac{i-1}{2}}, 0)$ and of an edge $e_{v_i\, v_{i+1}}$ with weight $(0, 2^{\frac{i-1}{2}})$; each vertex v_{i+1} is the start of an edge $e_{v_{i+1}\, v_{i+2}}$ with weight $(0, 0)$; there are no other edges. Let $t = 2^{\frac{n-1}{2}}$. Clearly, any path from v_1 to v_n takes either an edge $e_{v_i\, v_{i+2}}$ or instead both the edges $e_{v_i, v_{i+1}}$ and $e_{v_{i+1}\, v_{i+2}}$ for

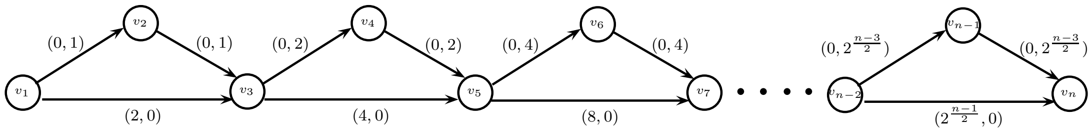


Figure 4.5: Alternative construction of Hansen's result without using zero weight edges.

$i = 1, \ldots, n-2$. Therefore, there are t paths in $\mathbf{G}$, and these paths have lengths $(t-1, 0), (t-2, 1), (t-3, 2), \ldots, (0, t-1)$. Hence, all paths of $\mathbf{G}$ are efficient paths for the bicriteria shortest path problem and constitute a minimal complete set. □

Technically, in Hansen's original proof he utilized zero weight edges $(0, 0)$ in the construction, but this is not really necessary as Figure 4.5 illustrates. A special construction of the coloured–edge graph based on the transformation introduced in Section 2.5 of Chapter 2 can be developed to prove the same result as Figure 4.6 shows.

The construction in Figure 4.6 is also capable of producing an exponential number of minimal paths. However, this section presents three constructions based on complete graphs with 3, 4 and 5 vertices that generate a number of minimal paths that equals the total number of simple paths in the graph. The total number of simple paths between two vertices in a complete graph is given by $\sum_{j=0}^{n-2} \binom{n-2}{j} j!$, Lawler [28]. The importance of these results is that a higher number of minimal paths might exist for biweighted graphs than that presented

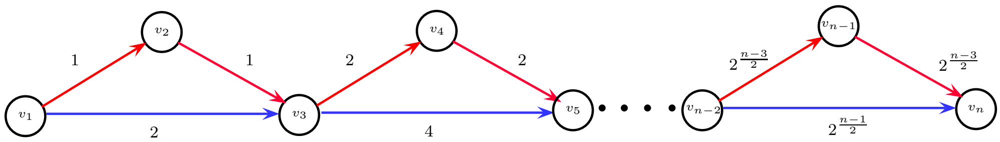


Figure 4.6: Special construction of the coloured–edge graph to prove Hansen's result.

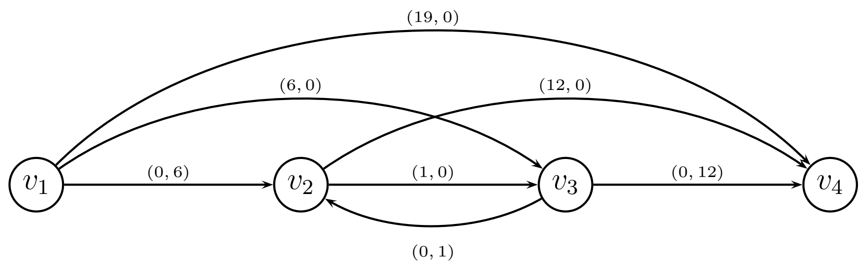


Figure 4.7: Biweighted graph whose five paths from v_1 to v_4 are minimal.

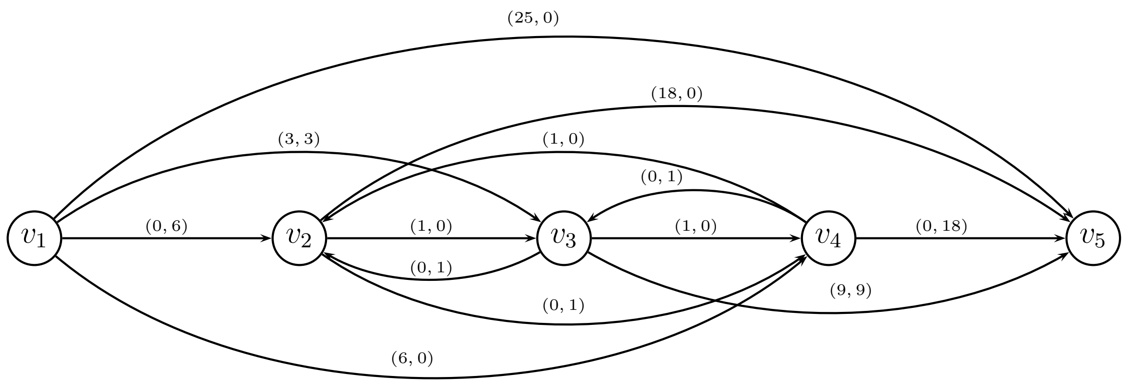


Figure 4.8: Construction of a biweighted complete graph whose 16 paths from v_1 to v_5 are minimal.

by Hansen.

To build a complete graph with three vertices in which all paths are incomparable is easy by just applying Hansen's construction. The four vertex case is a bit less obvious. Nevertheless, a construction of a graph with four vertices in which each of the five paths from v_1 to v_4 is incomparable is still attainable as Figure 4.7 shows.

The construction in Figure 4.7 requires the initial path v_1, v_2, v_3, v_4 to begin. Next more edges are added so that more incomparable paths are generated. This initially needs a trial and error process for the assignation of weight tuples capable of preserving incomparability. Similarly, a construction for a five vertices complete graph is also possible as Figure 4.8 shows. The total number of minimal

Table 4.2: Minimal paths in the five vertices biweighted complete graph.

Paths	Weights
v_1, v_2, v_5	$(18, 6)$
v_1, v_2, v_3, v_5	$(10, 15)$
v_1, v_2, v_3, v_4, v_5	$(2, 24)$
v_1, v_2, v_4, v_5	$(0, 25)$
v_1, v_2, v_4, v_3, v_5	$(9, 17)$
v_1, v_3, v_5	$(12, 12)$
v_1, v_3, v_4, v_5	$(4, 21)$
v_1, v_3, v_4, v_2, v_5	$(23, 3)$
v_1, v_3, v_2, v_5	$(21, 4)$
v_1, v_3, v_2, v_4, v_5	$(3, 23)$
v_1, v_4, v_5	$(6, 18)$
v_1, v_4, v_2, v_5	$(25, 0)$
v_1, v_4, v_2, v_3, v_5	$(17, 9)$
v_1, v_4, v_3, v_5	$(15, 10)$
v_1, v_4, v_3, v_2, v_5	$(24, 2)$
v_1, v_5	$(25, 0)$

paths as well as simple paths here is 16. Table 4.2 shows the corresponding set of incomparable paths.

Also there are constructions for $n = 6$ in which more than 50 of the 64 paths are minimal (so many more than can occur in a 2–coloured edge graph with $n = 6$ vertices). Whether weights can be chosen so that all 64 paths are minimal is currently unknown. However, from the four and five vertex cases it is seen that intermediate vertices are apparently following a pattern base on 0–1 tuples. Table 4.3 compares the maximum number of minimal paths obtained by the construction in a biweighted graph versus the maximum number of minimal paths of a bicoloured–edge graph.

The two presented cases pose an important question: For some family of graphs the number of minimal paths might be a factorial function of n. If this is true, then Hansen's result is not the actual maximum number of minimal paths. The sense in determining a more intractable result is justified by the fact that a factorial number of minimal paths could truly ratify the intractability of

Table 4.3: Comparison of the maximum number of minimal paths.

Vertices	Minimal Paths	
	Biweighted Graph	Bicoloured Graph
3	2	4
4	5	8
5	16	16
6	> 32	32

the bicriteria shortest path problem. The justification in doing this concerns the applicability of the coloured–edge graph for the modelling of multicriteria combinatorial optimization problems. The exponential bound developed for the coloured–edge graph is tight and guarantees that in the worst case no more than k^{n-1} path are minimal. Hence, a more intractable result for the bicriteria shortest path problem could make the coloured–edge graph a useful modelling alternative for some multicriteria problems.

The generalization of the construction as well as the determination of a tighter upper bound for the bicriteria shortest path problem based on them remain as an open problem.

Chapter 5

Probabilistic Analysis of the Coloured–Edge Graph

Numerical results shown in Chapter 3 raise two questions involving the behaviour of the cardinality of $\mathcal{M}_{uv}$ in weighted coloured–edge graphs

1. Why does the computation of minimal paths perform well in randomly weighted coloured–edge graphs?
2. Why does the cardinality of $\mathcal{M}_{uv}$ for coloured–edge graphs with Euclidean weights decrease when randomness is added to edge lengths?

In the answers lies an important understanding about the computational tractability of the coloured–edge graph. This chapter attempts to find the answers by probabilistically studying the coloured–edge graph in different settings. First, some probabilistic techniques are presented.

5.1 Introduction

Probabilistic analysis is the employment of probability in the analysis of problems, Ross [68]. Commonly, probabilistic analysis is used to study the running time of algorithms. Nonetheless, other quantities can also be analyzed such as the number of minimal paths in a coloured–edge graph. A probabilistic analysis is carried out by making assumptions about the distribution of the input. Then, the problem is analyzed by estimating the expected value of the variable under study, Cormen et al. [61]. Two techniques are popular for the probabilistic analysis of problems: Average Case Analysis and Smoothed Analysis.

Average Case Analysis focuses on the determination of the expected value of a complexity measure of a problem whose inputs are randomly generated. The usefulness of this technique is the mathematical rigor in explaining why some problems may be very tractable in practice despite their worst cases being theoretically intractable. Nevertheless, the main drawback of this technique is the interpretation of a random input. Random input may be unconvincing in practical situations as input parameters found in many applications can significantly differ from pure random numbers. Cormen et al. present the same idea differently. For some problems, it is reasonable to assume something about the distribution of all potential inputs so that average case analysis can be used as a technique for gaining insight into the problem. For other problems however, an input distribution cannot be described and in consequence an average is not useful. A drawback from a heuristic's design viewpoint is given by Simchi–Levi et al. [69], an average–case analysis is usually only possible for a heuristic that is very close to the optimal solution. This "closeness" is achieved in large instances of the problem. This implies that the analysis can only provide asymptotics of the complexity measure.

How likely in practice is it that our model becomes intractable? Theoretically, this question can be answered by performing a smoothed analysis on the model. Smoothed analysis was introduced by Spielman & Teng [70] as a hybrid between worst and average case analysis. This technique builds a probabilistic argument by adding slight perturbations to parameters in a model according to a probability distribution. The aim is to bound the probability that a current worst case solution(s) becomes more tractable as a result of the perturbations. In other words, smoothed analysis can be regarded as an interpolation between a worst case scenario and an average case scenario. Manthey & Reischuk [71] formally define smoothed analysis as follows: Let C be some complexity measure and f a probability distribution. The expresion $x \sim f$ is used to indicate that a variable x is randomly generated from f. The worst case complexity is $\max_x C(x)$, and the average case complexity is $\mathbb{E}_{x \sim f} C(x)$, where $\mathbb{E}$ denotes expectation with respect to a probability distribution f and x is drawn ($\sim$) according to f. The smoothed complexity is defined as $\max_x \mathbb{E}_{y \sim f(x,\sigma)} C(y)$. Here, x is chosen by an adversary and y is randomly chosen according to some probability distribution $f(x, \sigma)$ that depends on x and a parameter σ. Usually, $f(x, \sigma)$ favours y in the vicinity of x. The smoothing parameter σ, which is for instance the variance of $f(x, \sigma)$, denotes how strongly x is perturbed. Intuitively, for $\sigma = 0$, smoothed complexity becomes worst case complexity, while for large σ, smoothed complexity becomes average case complexity.

In simple words, if the smoothed complexity of an algorithm is low, then it is unlikely to accidentally come across an instance for which the algorithm behaves poorly, even if the worst case complexity of the algorithm is bad. In this case, worst case scenarios are isolated events.

A crucial feature of this technique is the construction of a perturbation model by which a random variable is added to the model's parameters. Particularly, the perturbation model employed by Spielman & Teng requires the definition of a very specific problem instance (such as the worst case) which is assumedly specified by an adversary. Next, the parameters of the instance are perturbed by adding a random number. This perturbation approach is known as the *two–step model*. However, this approach is restricted to both a continuous perturbation model and a Gaussian distributed random variable. It is important that the selected model make "sense" in the context of the problem. For example, Spielman & Teng indicate in [72] that a natural perturbation of a graph is obtained by adding edges between unconnected vertices and removing edges with some probability. However, a graph subject to such perturbations is highly unlikely to have a large clique (an embedded subgraph that is complete), and thus it could be meaningless to measure the performance of algorithms that determine the maximum clique. As a result, the performance of an algorithm under this perturbation model would be misleading since it is unable to establish instances close to the largest clique. In other words, a perturbation model must always maintain the original structure of a problem.

A more general model based on a semi–random input model was introduced by Beier & Vöcking [73]. In the *one step model* the adversarial instance is perturbed by adding random numbers to the inputs drawn according to a specified family of probability density functions satisfying the following conditions. Let $f : \mathbb{R} \to \mathbb{R}^+$ be a probability density function such that $\sup_s(f(s)) = 1$ and $\mathbb{E} = \int_{\mathbb{R}} |s|f(s)ds$ is finite. This means that the random variable described by f has a probability density function bounded by 1 and a finite expected value. The function f is a "perturbation model". Let f_ϕ be the scaled version of f. This means for $\phi \geq 1$ and every $s \in \mathbb{R}$, $f_\phi(s) = \phi f(s\phi)$. Thus, $\sup_{s\in\mathbb{R}}(f_\phi(s)) = \phi$ and the expected value is $\int_{\mathbb{R}} |s|f_\phi(s)ds = \mathbb{E}/\phi$. Therefore, perturbations according to the model f can be obtained by adding independent random variables with probability density function f_ϕ to each parameter of the adversary's instance. For example, Gaussian distributed perturbations can be obtained by setting f to be a Gaussian probability density function with standard deviation $(2\pi)^{-1/2}$. This probabilistic model is complemented by the following lemma that allows the determination of tail bounds for independent random variables.

Probabilistic analysis is used in this chapter to study the complexity of the coloured–edge graphs with random inputs. In particular, this chapter focuses on the determination of bounds on the number of optimal paths for coloured–edge graphs with a probabilistic input model.

5.2 Coloured–Edge Chains

The focus now is to establish the behaviour of the number of optimal paths in a coloured–edge chain with probabilistic inputs. The main motivation for carrying this out is concerned with the behaviour of the worst case. Chapter 4 provides an intractable example based on a coloured–edge chain. It could make sense at first glance to think that this example is not typically part of a real networked system.

To begin, pathological weighted coloured–edge chains are analyzed.

Lemma 5.2.1. *Let* $\mathbf{G}$ *be a complete chain with* k *colours and* n *vertices for which the weights of the edge* e_c *from* v_i *to* v_{i+1} *satisfy the following conditions:*

1. *For all colours* c*,* c'*, the edges* e_c *and* $e_{c'}$ *from* v_i *to* v_{i+1} *have the same weight,* $\omega(e_c) = \omega(e_{c'}) = \omega_i$*.*

2. *For the weights* $W = \{\omega_1, \omega_2, \ldots, \omega_{n-1}\}$ *the sum function* $\sum : \mathcal{P}(W) \setminus \emptyset \to \mathbb{R}^+$ *is one to one. For instance, if* $\{\omega_{i_1}, \omega_{i_2}, \ldots, \omega_{i_s}\}$ *and* $\{\omega_{j_1}, \omega_{j_2}, \ldots, \omega_{j_t}\}$ *are two distinct non–empty subsets of* W*, then* $\omega_{i_1} + \omega_{i_2} + \cdots + \omega_{i_s} \neq \omega_{j_1} + \omega_{j_2} + \cdots + \omega_{j_t}$*.*

Then $\mathcal{M}_{v_1\, v_n}$ *has cardinality* k^{n-1}*.*

Proof. Firstly note that by condition 1, for any path p from $u = v_1$ to $v = v_n$ the sum of all the edge weights in the path is fixed:

$$\sum_{colour\ c} \omega_c(p) = \sum_{i=1}^{n-1} \omega_i.$$

Suppose p and q are two distinct paths from $u = v_1$ to $v = v_n$. Then in some colour c', p and q must use different edges, so by condition 2 $\omega_{c'}(p) \neq \omega_{c'}(q)$. Without loss of generality suppose $\omega_{c'}(p) < \omega_{c'}(q)$. As

$$\sum_{colour\ c} \omega_c(p) = \sum_{colour\ c} \omega_c(q)$$

it follows that there is another colour c'' for which $\omega_{c''}(p) > \omega_{c''}(q)$. Hence p and q are incomparable, so all k^{n-1} paths are minimal. □

Note that if the weights $\omega_1, \omega_2, \ldots, \omega_{n-1}$ are independent random variables then almost always the second condition of the theorem holds. Hereafter, a chain meeting the conditions of Lemma 5.2.1 is called a *uniformly weighted coloured–edge chain.* As a special case the exponentially weighted coloured–edge chain from Chapter 4 satisfies the conditions of the lemma.

Corollary 5.2.2. *Suppose* **G** *is a coloured–edge chain with* n *vertices and* k *colours that has Euclidean weights in* $\mathbb{R}$ *where the vertices are randomly positioned. Then* **G** *has* k^{n-1} *minimal paths.*

A weighted coloured–edge chain in two colours is next analyzed. The following lemma demonstrates that some chains can instead be very well behaved.

Lemma 5.2.3. *Suppose* **G** *is a bicoloured–edge chain with* n *vertices* $v_1, v_2, \ldots, v_n$ *whose* $n-1$ *edges* $e_i : v_i \to v_{i+1}$ *in colour* c *satisfy*

$$\omega(e_1) < \omega(e_2) < \ldots < \omega(e_{n-1})$$

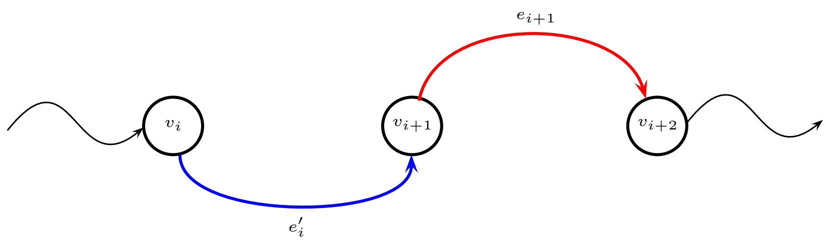


Figure 5.1: Argument used for proving Lemma 5.2.3.

and in colour c' satisfy

$$\omega(e'_1) > \omega(e'_2) > \ldots > \omega(e'_{n-1}).$$

Then $\mathcal{M}_{v_1\ v_n}$ has cardinality n.

Proof. Firstly, consider any path p from v_1 to v_n that for some $i \leq n-2$ uses edges e'_i and e_{i+1} (see Figure 5.1). As $\omega(e'_i) > \omega(e'_{i+1})$ in colour c' and $\omega(e_{i+1}) > \omega(e_i)$ in colour c, p has weight $(\omega_c(p), \omega_{c'}(p))$ which is greater than the weight of the path q that uses the same edges as p for $j < i$ and for $j > i+1$, but uses edges e_i and e'_{i+1} in place of e'_i and e_{i+1}. Hence, p is not minimal. Consequentially, any minimal path can only use edges $e_1, \ldots, e_i, e'_{i+1}, e'_{i+2}, \ldots, e'_{n-1}$ for some $0 \leq i \leq n-1$, of which there are n possible paths which are easily seen to be incomparable. □

Corollary 5.2.4. *Suppose* **G** *is a bicoloured–edge chain with n vertices, whose $2(n-1)$ edge weights are independent random variables. Then the probability that $\mathcal{M}_{v_1\ v_n}$ has cardinality n is at least $1/(n-1)!$*

Proof. Without loss of generality it can be presumed that $\omega(e_1) < \omega(e_2) < \ldots < \omega(e_{n-1})$ as the vertices can be rearranged in any chain without affecting any path weights. Then the weights $\omega(e'_1), \omega(e'_2), \ldots, \omega(e'_{n-1})$ can be arranged in $(n-1)!$ ways, all equally likely and one arrangement satisfies the criteria of the previous lemma. □

5.3 Expected number of minimal paths

This section seeks to demonstrate that the expected number of minimal paths for a bicoloured–edge graph is polynomially bounded. The main approach is based on the works of Röglin & Vöcking [38] and Beier et al. [74]. However, several arguments have been modified to be applied in the context of coloured–edge graphs.

Suppose $\mathbf{G}$ is a finite weighted bicoloured–edge graph with colours $\{red, green\}$ for convenience and let u, v be vertices of $\mathbf{G}$ for which there is a pure coloured–edge path in colour *green* from u to v.

Assume e is a *red* edge of $\mathbf{G}$ whose weight is a random variable with bounded probability density function $f_e : (0, \infty) \to [0, \phi_e]$ for some $\phi_e > 0$.

Define the function $\Delta_e : [0, \infty) \to (0, \infty]$ for $r \geq 0$ by the following. Consider the paths p_e from u to v that do not include the edge e and for which $\omega_{red}(p_e) \leq r$. Since there is a pure path in colour *green* from u to v, there are such paths p_e, and since $\mathbf{G}$ is finite there are only finitely many such paths. Take p_e^{max} to be such a path that has least *green* weight $g_r = \omega_{green}(p_e^{max})$. Note that g_r is uniquely defined for r and does not depend in any way on the value of $\omega(e)$.

Next, consider the paths q_e from u to v that include the edge e and for which $\omega_{green}(q_e) < g_r$. If there is no such path then take $\Delta_e(r) = \infty$ for convenience, otherwise let q_e^{min} denote such a path that has least *red* weight and take $\Delta_e(r) = \omega_{red}(q_e^{min})$. Note that although $\omega_{red}(q_e^{min})$ depends on the value of $\omega(e)$, the weight of this edge does not affect the relative *red* ordering between the various q_e (since they each include e). Hence the choice of q_e^{min} (or another q_e with same *red* weight) does not depend in any way on the value of $\omega(e)$, and $s_r = \omega_{red}(q_e^{min}) - \omega(e)$ (where s_r is the sum of *red* weights except for e) is uniquely determined by r and does not depend on the choice of $\omega(e)$.

Figure 5.2 illustrates Δ_e and its associated variables, black dots are used to represent paths not using edge e, whereas white dots represent otherwise. Note that all q_e paths just shift horizontally depending on the value of $\omega(e)$. However, they do not change their relative positions.

Lemma 5.3.1. *For any $r \geq 0$ and $\varepsilon \geq 0$, if $\Delta_e(r) < \infty$ then*

$$\mathbb{P}(r < \Delta_e(r) \leq r + \varepsilon) \leq \phi_e \cdot \varepsilon.$$

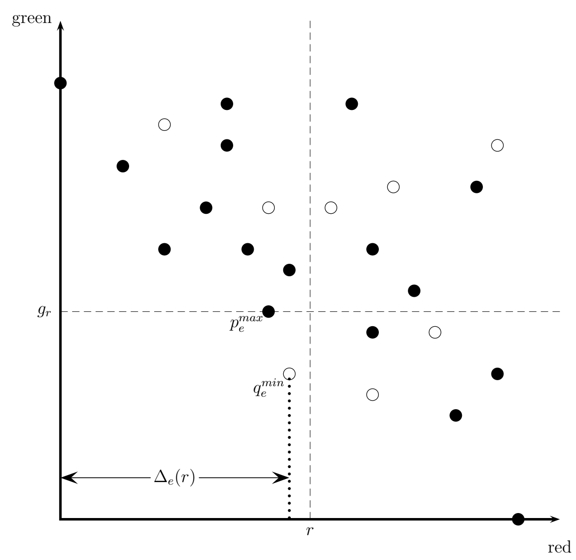


Figure 5.2: Representation of $\Delta_e(r)$ and associated variables.

Proof. Let $r \geq 0$, $\varepsilon > 0$ and q_e^{min} be a path that includes e with $\omega_{green}(q_e^{min}) < g_r$ and $\omega_{red}(q_e^{min})$ minimal amongst such paths. As s_r does not depend on the value of $\omega(e)$

$$\begin{aligned}
\mathbb{P}(r < \Delta_e(r) \leq r + \varepsilon) &= \mathbb{P}(r < s_r + \omega(e) \leq r + \varepsilon) \\
&= \mathbb{P}(r - s_r < \omega(e) \leq r - s_r + \varepsilon) \\
&= \int_{r-s_r}^{r-s_r+\varepsilon} f_e(x)dx \\
&\leq \int_{r-s_r}^{r-s_r+\varepsilon} \phi_e dx = \phi_e \cdot \varepsilon.
\end{aligned}$$

□

Now, suppose for the graph **G** that all its *red* edges e have weights that are random variables with bounded probability density functions, and suppose there is also a pure coloured–edge path in *red* from u to v, with a minimal pure coloured–edge path in *red* having *red* weight r_{tot}.

Define the function $\Delta : [0, r_{tot}) \to (0, \infty)$ for $0 \leq r < r_{tot}$ by the following. Consider the minimal paths q from u to v for which $\omega_{red}(q) > r$. Since there is a pure red path with weight r_{tot}, there are such paths q, and since **G** is finite

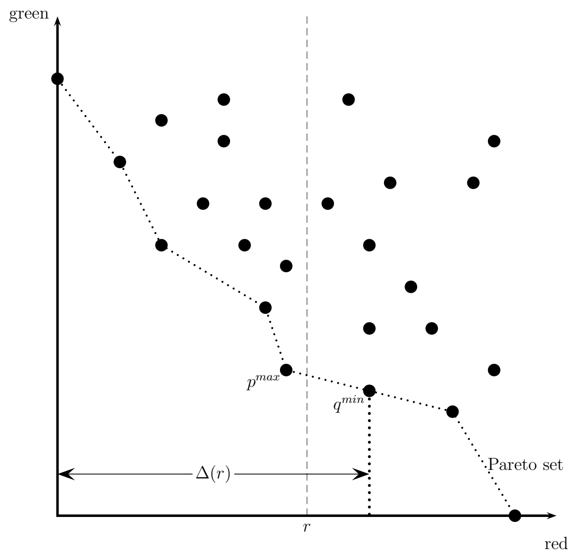


Figure 5.3: Representation of $\Delta(r)$ and associated variables.

there are only finitely many such paths. Take q^{min} to be a minimal path with $\omega_{red}(q^{min}) > r$ that has least red weight and take $\Delta(r) = \omega_{red}(q^{min}) > r$. Figure 5.3 illustrates $\Delta(r)$ and q^{min}.

Lemma 5.3.2. *For $0 \leq r < r_{tot}$, there exists a red edge e for which $\Delta(r) = \Delta_e(r)$.*

Proof. Consider the minimal paths p from u to v for which $\omega_{red}(p) \leq r$. Since there is a pure *green* path from u to v there are such paths, and since **G** is finite there are only finitely many such paths.
Take p^{max} to be a minimal path with $\omega_{red}(p) \leq r$ that has greatest *red* weight, so p^{max} and q^{min} are adjacent minimal paths in the Pareto set (see Definition 2.4.3). Note that there can be no path for which both its *red* weight is less than $\omega_{red}(q^{min})$ and its *green* weight is less than $\omega_{green}(p^{max})$. As a result, there cannot be minimal paths between p^{max} and q^{min} (Figure 5.4). Since $\omega_{red}(p^{max}) \leq r < \omega_{red}(q^{min})$ there must be some *red* edge e that is in q^{min} but not in p^{max}.
As p_e^{max} has the least *green* weight amongst paths p_e that do not include e and for which $\omega_{red}(p_e) \leq r$, $\omega_{green}(p_e^{max}) \leq \omega_{green}(p^{max})$. However, since $\omega_{red}(p_e^{max}) \leq r < \omega_{red}(q^{min})$ and there are no paths between p^{max} and q^{min} it follows that $\omega_{green}(p_e^{max}) \geq \omega_{green}(p^{max})$. Hence $g_r = \omega_{green}(p^{max})$. As p^{max} and q^{min} are incomparable $\omega_{green}(p^{max}) > \omega_{green}(q^{min})$. Next, as q_e^{min} has the least *red* weight amongst paths q_e that do include e and for which $\omega_{green}(q_e) < g_r$, $\omega_{red}(q_e^{min}) \leq$

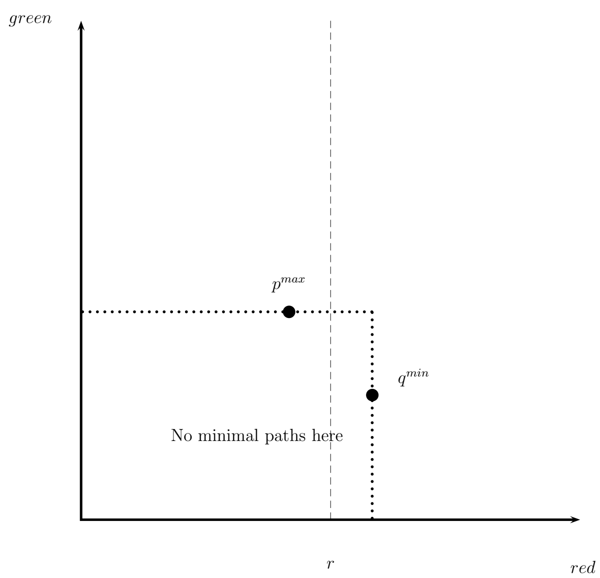


Figure 5.4: Argument used to prove Lemma 5.3.2.

$\omega_{red}(q^{min})$. But since $\omega_{green}(q_e^{min}) < g_r = \omega_{green}(p^{max})$ and there are no paths between p^{max} and q^{min}, it follows that $\omega_{red}(q_e^{min}) \geq \omega_{red}(q^{min})$. Hence $\Delta(r) = \omega_{red}(q^{min}) = \omega_{red}(q_e^{min}) = \Delta_e(r)$. □

Corollary 5.3.3. *For any $0 \leq r < r_{tot}$ and $\varepsilon > 0$*

$$\mathbb{P}(\Delta(r) \leq r + \varepsilon) \leq \left(\sum_{red\ edge\ e} \phi_e \right) \cdot \varepsilon.$$

Proof. If $r < \Delta(r) \leq r + \varepsilon$ then by Lemma 5.3.2, there exists a *red* edge e for which $r < \Delta_e(r) \leq r + \varepsilon$. Hence using a union bound [1] and Lemma 5.3.1,

$$\begin{aligned} \mathbb{P}(r < \Delta(r) \leq r + \varepsilon) &\leq \sum_{red\ edge\ e} \mathbb{P}(r < \Delta(r) \leq r + \varepsilon) \\ &\leq \sum_{red\ edge\ e} \phi_e \cdot \varepsilon. \end{aligned}$$

□

Lemma 5.3.1 and Lemma 5.3.2 provide the main arguments to establish a bound on the expected number of minimal paths for a bicoloured–edge graph.

[1] A union bound (also known as Boole's Inequality) states that for any finite or countable set of events, the probability that at least one of the events happens is no greater than the sum of the probabilities of the individual events, Ross [68].

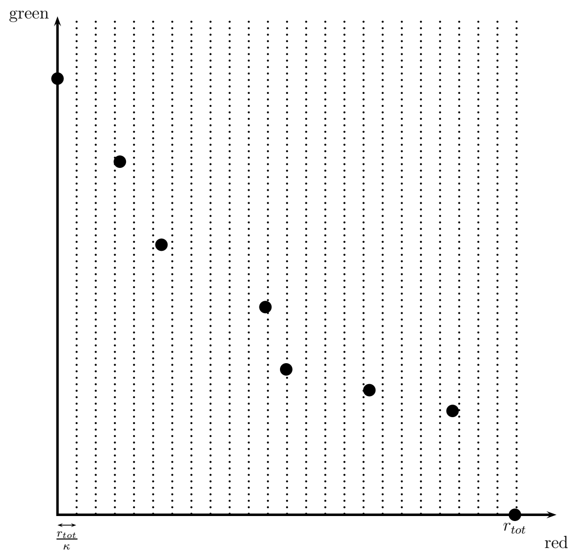


Figure 5.5: Pareto minimal elements and partition of the interval $(0, r_{tot}]$.

Theorem 5.3.4. *Let* $\mathbf{G}$ *be a finite weighted bicoloured–edge graph and let* u, v *be vertices of* $\mathbf{G}$ *for which there is a pure colour path from* u *to* v *in each of the two colours. Suppose that the weights of all edges* e *in one of the colours are random variables with probability density functions bounded above by* ϕ_e*, and let* r_{tot} *denote the weight of the minimal pure colour path in that colour. Then the expected number of Pareto minimal elements with distinct weights is bounded above by* $\sum_e \phi_e \cdot r_{tot} + 1$.

Proof. As previously denote the colours by $\{red, green\}$ for convenience. As $\mathbf{G}$ is finite there are only finitely many minimal paths q from u to v, and they all have *red* weight between 0 and r_{tot} inclusive.

Partition the interval $(0, r_{tot}]$ into κ equal subintervals and note that since only minimal paths with distinct (*red*) weights are considered, there is a threshold κ_{min} above which each interval can contain at most one minimal path. Hence for all $\kappa \geq \kappa_{tot}$, the expected number of distinct minimal paths is

$$1 + \sum_{i=0}^{\kappa-1} \mathbb{P}\left(\exists \text{ minimal path } q \text{ with } \frac{r_{tot}}{\kappa} i < \omega_{red}(q) \leq \frac{r_{tot}}{\kappa}(i+1)\right)$$

including the minimal pure *green* path that has *red* weight 0. Figure 5.5 depicts the partition of $(0, r_{tot}]$.

Now for each i there is a minimal path q with $\frac{r_{tot}}{\kappa} i < \omega_{red}(q) \leq \frac{r_{tot}}{\kappa}(i+1)$ if and only if $\Delta(\frac{r_{tot}}{\kappa}i) \leq \frac{r_{tot}}{\kappa}(i+1)$ which has probability bounded above by $(\sum \phi_e)\frac{r_{tot}}{\kappa}$ by Corollary 5.3.3. Hence the expected number of minimal paths with distinct weights is bounded by

$$1 + \sum_{i=0}^{\kappa-1} \left(\sum \phi_e\right) \frac{r_{tot}}{\kappa} = 1 + (\sum \phi_e) \cdot r_{tot}.$$

□

Note that if each *red* edge has weight bounded by r_{max} then r_{tot} is $O(n)$, so the expected number of minimal paths is $O(mn) + 1$ where m denotes the number of red edges. Furthermore, there are only at most $n^2 - 3n + 3$ red edges to consider in a graph with n vertices so the order is bounded by $O(n^3)$.

Theorem 5.3.4 indicates that bicoloured–edge graphs are typically tractable. The bound is also consistent with the experiments presented in Chapter 3 were several graphs with random weights were tested.

5.4 Bounds in Multicriteria Optimization

In their paper Röglin & Teng [75] obtained a bound for the number of Pareto optimal solutions for multicriteria optimization problems. Their bound was $O(n^{2^{d-1}(d+1)!})$,where d stands for the number of criteria. Their approach basically considered the bound of Beier et al. [74] as a base case in the construction of an inductive argument upon the number of criteria. Very recently, Moitra & O'Donnell [76] improved such bound to $O(n^{2d})$ by using a rather intricate family of probabilistic events. An open question proposed by Moitra & O'Donnell in their paper was about how much further their bound can be improved. Brunsch & Röglin [77] provided a pessimistic answer to this question in the sense that no significant improvement is attainable. They proved this by creating a very specific class of instances lower bounded by $\Omega(n\phi)^{(d-\log(d)\cdot(1-\Theta(1/\phi))}$.

Because of the recentness of these results, it is still unclear whether such constructions can be replicated in the context of the coloured–edge graph.

Chapter 6

The Coloured–Edge Graph in Practice

This Chapter experimentally studies the applicability of the coloured–edge graph as a modelling tool in real multimodal networks. To do this, several datasets have been constructed. These datasets describe real multimodal transportation systems of different types. The main idea is to utilize such dataset information as input for the multimodal Dijkstra's algorithm presented in Chapter 2. Three cases are studied. Firstly, a comparison between multimodal transportation systems of New Zealand and Europe is carried out. Secondly, the limits of the current modelling approach are explored by a large dataset based on the multimodal transportation system of France. Finally, a dataset containing airlines and their corresponding worldwide routes is used by the multimodal Dijkstra's algorithm for computing minimal flight routes. Results in this chapter support the theoretical finding developed in previous sections.

6.1 Introduction

Multimodal networks arise in a variety of applications. Particularly freight and urban transportation are application domains of interest because of their importance nowadays in business. These two application domains also account for a considerable number of applied studies about multimodal networks.

This chapter is devoted to investigating real application cases of multimodal networks in the context of transportation. A series of tests in this chapter based on real transport networks seeks to assess the performance of the coloured–edge graph approach in real as well as large multimodal datasets.

The importance in assessing the performance of any approach in a realistic instance stems from the fact that these instances behave very differently to those generated in a random fashion. As Zhan & Noon indicate [78] most computational evaluations in the literature take random networks as input. These networks usually range from complete networks with uniformly distributed arc lengths to highly structured grid networks. However, such experimental inputs often differ from real cases in both the degree and configuration of the network's connectivity. Moreover, random networks can induce irregularities that might considerably favor certain types of algorithms and drastically disfavor others.

Large multimodal datasets are typical in the real world. This justifies the need for an assessment under this circumstance. Largeness is a variable that in real scenarios can be considered by taking the size of previous research into consideration. As far as this thesis is concerned, a large multimodal network can be regarded as such in terms of either its number of vertices and edges or instead transport modes. The latter means the use of a higher number of colours in a coloured–edge graph.

Two works in particular raise attention regarding the analysis of real multimodal networks. This is mainly because of the size of their tested instances. In their paper Ayed et al. [79] applied a hybrid approach for solving the time–dependent multimodal transportation problem. The approach is based on a transfer graph model that is optimized by an ant–colony heuristic. Both model and optimization technique are applied on realistic instances of a multimodal network. The author reported results for datasets based on real urban multimodal networks from several cities around the world. The largest analyzed network was composed of 4000 vertices, 15000 edges and 5 transport modes. The only pitfall with this dataset is that it is an "equivalent" instance of another paper's network. This means the authors emulated a dataset from another paper by generating a network that accordingly follows parameters such as the number of vertices and edges, modes and network density. A second appealing paper is provided by Von

Ferber et al. [80]. These authors presented a practical study which evaluated statistical properties of fourteen urban multimodal transportation networks from several cities. The compiled datasets stand out in terms of their largeness. The largest multimodal dataset obtained corresponds to the city of Hamburg with 8084 vertices, 708 transport routes and 5 transport modes. The authors use the statistical properties of the networks to build a model that is capable of reproducing the majority of the features of urban transportation systems found in German cities. This model is mainly based on the dynamics of routes growth.

Three real cases are used in this chapter as input for the multimodal Dijkstra's algorithm. Case 1 explores five moderate multimodal transportation systems that come from New Zealand and some European countries. Cases 2 and 3 analyze the approach from two different perspectives of the size of the input. Case 2 emphasizes the number of vertices as variable determining the network size, whereas in Case 3 the number of transport modes (colours) is the predominant variable. All data employed in the following sections required a level of preprocessing so that they could be used as input to the multimodal Dijkstra's algorithm.

6.2 Case 1: Multimodal Systems of New Zealand and Europe

An appraisal is carried out in this section to analyze the behaviour of the number of minimal paths of two sets of multimodal transportation networks. One set is the New Zealand transportation system which consists of road, rail and airways. The other set consists of different transportation networks from four European countries. Railways and roadways are the dominant modes in this case. Apart from studying the number of minimal paths, another interest of this section is to establish whether European multimodal transportation systems differ from the New Zealand one. The modelling technique used the coloured–edge graph in such a way that modes, cities, intercity links and distances are modelled by colours, vertices, edges and weights respectively. The resulting graph is used by the multimodal Dijkstra's algorithm as input to compute the minimal Pareto set of paths from a specified source vertex to all other vertices in the graph.

Table 6.1: Characteristics of the Networks.

Network	Country	Vertices	Edges	Modes
1	Denmark	124	1284	Road,Rail
2	Hungary	305	7418	Road,Rail
3	Spain	901	5326	Road,Rail
4	Norway	122	641	Road,Rail,Airways
5	New Zealand	183	1436	Road,Rail,Airways

6.2.1 Networks setup

The study built one multimodal network for New Zealand and four for Europe. Vector data information about Denmark, Hungary, Spain, Norway and New Zealand were collected from a GIS library (Geofabrik [81]) for this end. Countries were selected based on the degree of similarity they have regarding shape and number of locations. For example, New Zealand has resemblances with Norway in shape and number of vertices. Both countries have an elongated shape and between 100 and 200 locations.

The multimodal networks were stored and maintained as a set of vertices and bidirectional links. A network dataset for each mode was generated by firstly snapping vertices (towns and cities) to network features according to a tolerance radius. Secondly, a connectivity map was created by an ad-hoc algorithm that iterates itself through vertices. As a spin–off, this algorithm also calculated the real intercity distances as decimal geographic degrees. Additionally, airways were added as a third mode for Norway and New Zealand. Straight distances between airports were used as edge length in this case and airports had to be snapped to cities to build a connectivity map. Airway data was obtained from OpenFlights [82]. Characteristics of the resulting networks are shown in Table 6.1.

As an illustration, Figure 6.1 displays the Hungary roadway system which is composed of motorways and primary roads. Likewise, Figure 6.2 yields a view of the New Zealand airway and road systems. For New Zealand, the connection between the north and the south island is made only by airplane.

The reported runtimes correspond to CPU times measured while computing

Figure 6.1: Hungary roadway system.

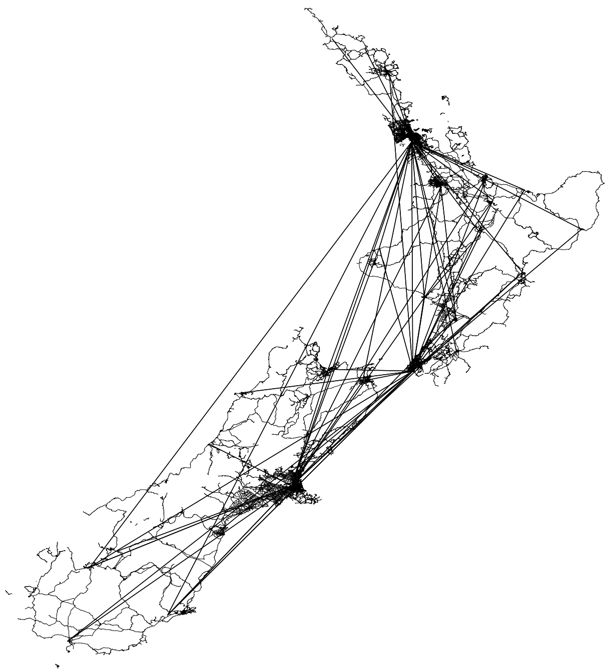


Figure 6.2: New Zealand airway and road system.

Table 6.2: Results for multimodal networks from vector data: Scenario 1.

Network	Source city	Average $\mathcal{M}_{uv}$ cardinality	Maximum $\mathcal{M}_{uv}$ cardinality	Processing Paths	CPU time
1	Copenhagen	56	171	10704	0.515
2	Budapest	69	227	53587	6.024
3	Madrid	133	1039	181976	32.216
4	Oslo	41	147	6248	0.158
5	Wellington	759	9342	611230	311.184

the total number of shortest paths ($\mathcal{M}_{uv}$ cardinality) from a source vertex to all other vertices. Networks were all tested on a standard desktop computer with dual core, 1.86 GHz CPU and 1.99 GB of RAM.

The $\mathcal{M}_{uv}$ cardinality for each city v (vertex) was calculated using two different source vertex scenarios. Scenario 1 considered the capital city of each country as the source vertex whereas Scenario 2 uses a city chosen at one end of the network as source. For instance, Wellington and Kaitaia were picked as source vertices for scenarios 1 and 2 respectively in the New Zealand case. In addition, the algorithm reported the total number of processing paths (total number of paths taken by the iterative subroutine of the algorithm) as well as average and maximum $\mathcal{M}_{uv}$ cardinality. Average cardinality was calculated by averaging all destination vertex cardinalities whereas the maximum cardinality corresponds to the largest value of $\mathcal{M}_{uv}$ cardinality among destination vertices.

6.2.2 Results

Results for Scenario 1 are shown in Table 6.2. CPU times are given in seconds. One fact that Table 6.2 demonstrates is that just a small fraction of paths are minimal in comparison to the number of processing paths taken by the algorithm. No more than 2% of such paths are found to be minimal for the studied multimodal networks. This is a promising result from a tractability viewpoint as the number of minimal paths that might need to be considered further is generally quite small.

Spain and New Zealand obtained the largest values of $\mathcal{M}_{uv}$ cardinality. What these countries have in common is a high level of network overlap between road and rail as well as a high number of cities located along such overlaps. These features together induce a high number of minimal paths because some network sections resemble coloured–edge chains. Coloured–edge chains are important subgraphs in a general coloured–edge graph because they are able to potentially result in a worst case scenario. Chapter 4 showed a construction of a coloured–edge chain that produced an exponential number of minimal paths. When this construction occurs in a coloured–edge graph, the total number of minimal paths is bounded above by k^{n-1}. In practical terms, those cities (or towns) that are reached via two or more intertwined modes are prone to generate an elevated number of shortest coloured–edge paths. To envisage the concept of network overlap, Figure 6.3 shows road and rail networks for New Zealand.

On the other hand, the rich variety of network links presented in Denmark, Hungary and Norway have less overlap so that the number of minimal paths tends to be lower. Moreover, maximal cardinalities were found for remote cities (or towns) with no direct link from the sources. For example, Frederiksharn and Rakamaz were the locations reporting the maximum number of minimal paths

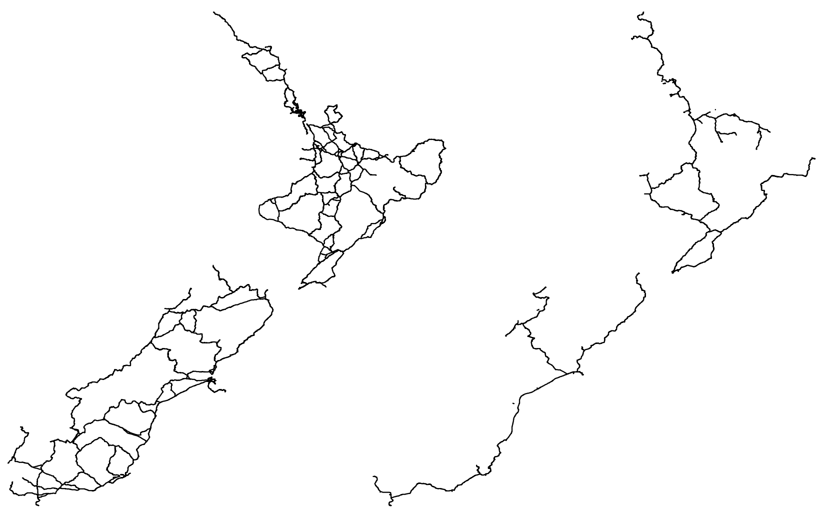


Figure 6.3: Network overlap between road (left) and rail (right) networks.

Table 6.3: Results for multimodal networks from vector data: Scenario 2.

Network	Source city	Average $\mathcal{M}_{uv}$ cardinality	Maximum $\mathcal{M}_{uv}$ cardinality	Processing Paths	CPU time
1	Hanstholm	76	332	20009	1.262
2	Csenger	149	600	135218	68.423
3	Tarroella	423	1864	606609	507.657
4	Ergersund	78	245	11674	0.332
5	Kaitaia	5969	33246	1644768	20572.52

for Denmark and Hungary, respectively. This indicates the number of minimal paths is affected when the location of the destination is changed. A special case is a vertex located at one of the country extreme ends.

Scenario 2 was set to analyze the impact that source location has on the cardinality of $\mathcal{M}_{uv}$. Table 6.3 summarizes the corresponding results. The selected source locations were extreme points situated at one of the four cardinal points. Spain and New Zealand again concentrate the highest numbers of paths. Here, maximum cardinalities were obtained to destinations San Fernando (Spain) and Riverton (New Zealand). Long shapes again account for a high number of minimal paths.

6.2.3 Analysis

The cardinality of $\mathcal{M}_{uv}$ was investigated in this section for real multimodal transportation systems. Such cardinality turns out to be higher and significantly so for locations situated far away from the sources in those countries whose transportation systems exhibited a greater level of intertwining. Intertwining results that certain sections of a multimodal network may resemble a coloured–edge chain. These chains were proved to be a potential cause of an exponential number of minimal paths in a coloured–edge graph.

Computational times were reasonable considering that the multimodal Dijkstra's algorithm was implemented with a basic data structure (priority queue). Multimodal networks in Scenario 2 required longer runs than Scenario 1 due to the greater number of paths processed by the algorithm. This indicates higher number of minimal paths to reach distant cities (or towns) when the source vertex is located at the very extreme of a country. Scenario 2 demonstrates that country shapes can significantly alter the number of minimal paths. Longer and slimmer shapes are closer to behaving like coloured–edge chains where the edge weights are correlated between modes. Hereby, the special shape of New Zealand is also accounting for the elevated number of minimal path found for its extreme locations. Although Norway has a shape resemblance with New Zealand, the lower number of minimal paths is explained by its different circumstances: (1) Norway's rail system does not connect as much of the country. Rail roughly covers just 20% of the territory. (2) The multimodal transportation system of Norway goes from dense to very sparse as a user moves from the south to the north. Road is predominantly defining the connectivity in the north. (3) The number of airways is much lower than in New Zealand. New Zealand has about 116 different air connections whereas Norway has just 62.

A significant difference between Europe and New Zealand can be established by analyzing a post–optimal scenario for each. After selecting a minimal path from the final Pareto set based on some overall measure of cost, a natural question to pose is how easily the chosen path should change when the cost measure fluctuates. A minimal solution is understood as a unique solution picked from the final Pareto set using some overall cost function. New Zealand possesses a greater number of minimal paths than any of the four tested European countries so that more alternatives are available for interchanging. This is because its two main transport modes present high level of intertwining (the train system tends to closely follows the road system producing similar path weights). On the other hand, Europe reports lower cardinalities due to a more "compact" shape of its networks. An optimal solution in this scenario can in general be harder to replace by small fluctuations in an overall cost function. In summary, a selected path can be expected to better cope with perturbations on relative costs in Europe rather than New Zealand. Expressed in a different way, a cost function that depends on distance in each mode would typically require greater variations on its coefficients in case of Europe to effect a change in the best path.

Table 6.4: ArcGIS data for France multimodal network.

Modes	Number of Junctions	Number of polylines	Polyline length		
			Maximum	Mean	Stnd. Dev.
Roadways	53562	47660	0.868028	0.010674	0.032452
Railways	18671	20083	1.280264	0.014966	0.046192
Motorway	7720	7432	1.221951	0.033488	0.078485
Waterways	17113	11635	3.238686	0.032070	0.095573

6.3 Case 2: France Multimodal Network

The approach is now tested on a large multimodal network. In this test, largeness is in the sense of number of vertices and edges. The selected network scenario corresponds to the multimodal transportation system of France being one of the largest networks in Europe. The multimodal network was obtained from vector data information retrieved from a public GIS library, Geofabrik [81].

The network dataset for each transport choice was firstly processed in ArcGIS to make it suitable for computation. ArcGIS is a Windows platform application for the analysis and processing of vector geographic information system data. This application has a network analysis extension that permits the identification of junctions and polylines [1] in each transport system. In addition, ArcGIS also has a macro for the computation of the adjacency matrix for each system of junctions

Table 6.4 summarizes the number of junctions and polylines given by ArcGIS for each transport mode as well as some statistics of the networks. All edge lengths are given in decimal geographic degrees. Four transport modes comprise the France transport system: road, rail, waterways and motorways. The road system mainly consists of primary roads but not roads classified as secondary. The rail system is comprised of common train lines disregarding subway and tram. Waterways are the channels and rivers used as transportation links. Finally, the motorway system of France includes toll roads and is considered a different mode

[1] In computer graphics a "polyline" is also known as "polygonal chain" which is a connected series of line segments, Burke [83].

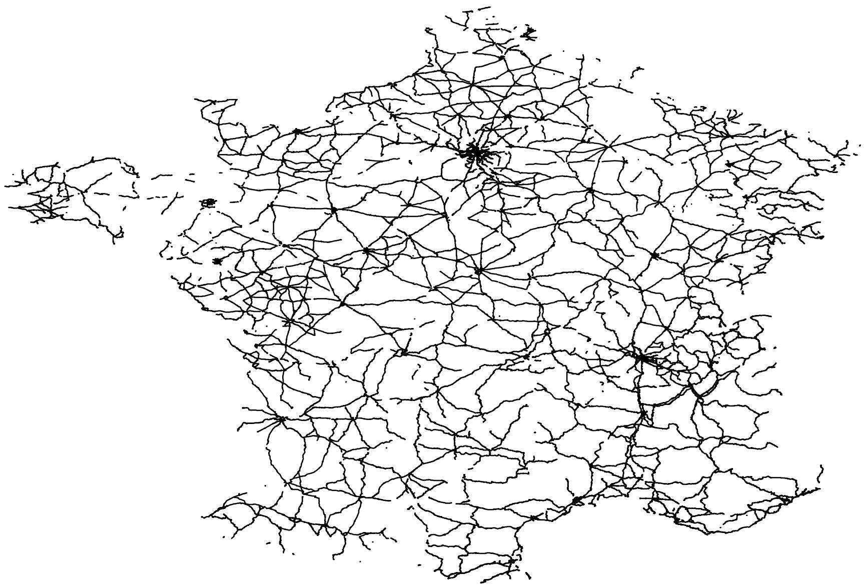


Figure 6.4: France roadway system.

of transport in its own right. As an illustration, Figure 6.4 depicts the France road system.

The construction of the multimodal network requires assembling the data for the four network modes together. This task is accomplished by an ad–hoc algorithm coded in the Java language. The code basically takes two inputs. These are the adjacency matrix of each transport mode network and a list of minimum interjunction distances in each mode. The latter is built in ArcGIS by taking a mode junction dataset and applying the "join and relates" tool with respect to each other mode junction dataset. This information facilitates the performance of a subsequent clustering procedure used inside of the ad–hoc algorithm. As ArcGIS treats the data for each transport mode as a separate network the clustering ensures that junctions that are within a specified radius of each other are identified to allow transfer between different modes. A partial view of the corresponding input files is found in Appendix A.

Two parameters need to be specified once the algorithm code is executed. A minimum clustering distance (this generates the network vertices) and a source vertex (a junction number). After entering this information, the Java code in-

Table 6.5: Results of Case 2.

Cluster	Number of Vertices	Number of Edges	Running Time	Avg. Paths	Max Paths
0.150	1501	4216	0.0180	33.0930	702
0.140	1869	5218	0.6950	208.270	3320
0.130	2343	6280	4.4010	404.380	14972
0.120	2948	7696	245.40	2013.82	37128
0.115	3108	8018	823.10	3575.03	46984

vokes the multimodal Dijkstra's algorithm, reporting upon completion a list with the total number of minimal paths to each vertex together with two additional variables: the maximum number of paths found in a particular vertex and the average number of paths.

This dataset was tested by firstly clustering junctions between 0.150 and 0.115 decimal degrees (14 to 11 km). The resulting networks together with running times (minutes) and average number of minimal paths are shown in Table 6.5. The computations were performed on a standard desktop computer with dual core 2.93 GHz CPU and 8 GB of RAM that was set with the queue version of the multimodal Dijkstra algorithm.

The results can be compared with some large multimodal networks from the literature. Comparability is established here in terms of number of vertices and edges. Networks from references [20], [84], [85] and [86] are used, which appear to be among the largest practical applications found in journal articles about multimodal networks. Unfortunately, these papers do not provide enough information to fully replicate the experiments. In addition, the data are not longer available from the authors. Technical features of the networks as well as the corresponding computational times (minutes) cited in the papers are shown in Table 6.6. The times given in the table should be interpreted with caution as they have different goals and probably utilize different computer hardware specifications.

Mathematical programming is applied in [20] to optimize a large international intermodal system considering multi–commodity options. In [84], a urban multimodal travel system is designed based on object modelling techniques and a

Table 6.6: Large networks from the literature.

Network	Reference	Nodes	Edges	Modes	Running time
1	[20]	112	407	5	2.00
2	[84]	1000	2830	5	0.11
3	[85]	1000	2747	2	0.23
4	[86]	3488	7200	2	5.00

K–shortest path algorithm. An algorithm for a time-dependent multimodal network is constructed in [85] based on a label-setting technique which invokes an optimality equation to update distance labels. Lastly, [86] studies a long–haul freight transportation system which is optimized by a heuristic approach based on a multicriteria shortest path problem model. All models are applied in the context of the constrained shortest path problem, and none are multigraphs unlike the France dataset.

As seen, the France multimodal network is as large as those instances found in the literature. Despite France's networks required longer runs of the multimodal Dijkstra's algorithm when the clustering distance is reduced, it cannot be disregarded that no constraints or reductions were required for obtaining the results in Table 6.5.

6.4 Case 3: World–Wide Airline Routes

The third experimental group is a worldwide dataset containing airline routes. The data was obtained from OpenFlights [82] and comprises of over 60000 world–wide flight connections for 655 international airlines together with the location of 5400 main global airports. For instance, some of the airlines included in the dataset are Delta (2400 connections), United Airlines (2200 connections) and American Airlines (1244 connections). These companies together account for 5844 coloured edges in the entire traffic airline system. These airlines can therefore be regarded as different transportation modes of a multimodal network. All such data is formatted as a text file to be read by a Java program. This program firstly filters the flight routes according to the name of the airlines and airport

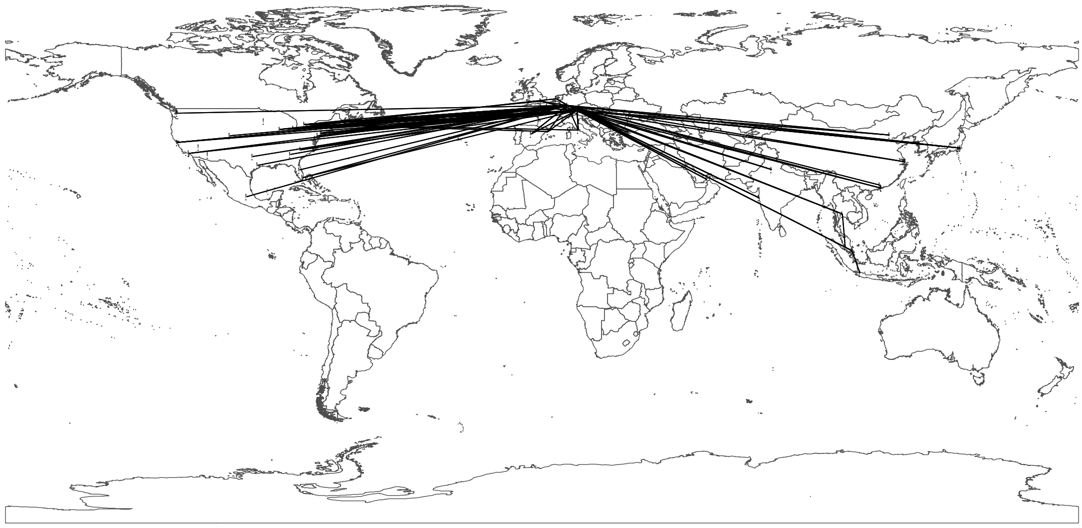


Figure 6.5: Lufthansa routes among the top 50 international airports.

names and next prepares the data in such a way that it can be used as input to the Multimodal Dijkstra's algorithm. Thus, the main inputs to this program are a set of airlines and their routes, a list of airports and their locations and a source airport.

The airline data allows the coloured–edge graph modelling approach to be evaluated in terms of a higher number of colours rather than vertices. Based on this, a list containing the top 50 international airports in terms of passenger traffic and a top 10 ranking of the 10 most popular international airlines are utilized as input to the algorithm. Direct arc distance in kilometers was considered as edge weights between airports. Figure 6.5 illustrates the routes of Lufthansa Airlines as an extract of the multimodal network. Additionally, a partial view of the data (as text format) is found in Appendix B.

The test was performed by increasing the number of airlines one at a time. The two top airlines, Delta and United Airlines, were initially selected from the ranking. Next, the other five were included one by one so that the number of colours was gradually increased.

The DAG data structure was tested against the queue implementation in or-

Table 6.7: Results of Case 3.

Num Airlines	Number of Vertices	Number of Edges	Max Paths	Avg Paths	Queue R.Time	DAG R.Time
2	50	811	57	11.84	0.07	0.23
3	50	865	573	72.12	0.28	0.75
4	50	1138	10262	983.54	30.64	1857.28
5	50	1310	22843	3586.18	701.94	86749.83
6	50	1388	125952	18577.62	16939.56	191642.38
7	50	1390	125645	18674.70	17387.20	201543.41

der to assesses its performance in a real multimodal network. The main goal was to verify whether the performances seen in random graphs repeats itself. Final results are shown in Table 6.7. Times are stated in seconds. The computations were performed on a standard desktop computer with dual core 2.93 GHz CPU and 8 GB of RAM that was set with both the queue and DAG versions of the multimodal Dijkstra algorithm. The computational capacity just allowed to increment the number of airlines up to 7 (Delta, United Airlines, Southwest Airlines, American Airlines, Lufthansa, China Southern and Ryanair).

Results show a significantly better performance of the queue data structure over the DAG. This fact supports the experimental results presented in Chapter 3 where these data structures were tested on random instances of the coloured–edge graph. Clearly, the number of airlines (colours) is the determining factor of the tractability. Results also reveal a high number of minimal paths to Jakarta International Airport in the case of 6 and 7 colours. This mostly is a product of the lack of direct connections from European and Western countries. Thus, most of the flights between distant locations have to traverse several intermediate airports. This fact is also explained by the low number of connections of the seventh airline in the ranking (Ryanair). This airline carries a significant number of passengers but has low connectivity among the 50 airports. This also accounts for the similarity of the running times for the queue data structure between the 6 and 7 number of airlines.

Chapter 7

Conclusions and Future Work

7.1 Multimodal Networks

Multimodal networks are pervasive. An extensive number of real systems can be modelled by these networks. As a modelling tool, multimodal networks seem to provide a straightforward approach for the representation of natural systems where several modes of transport operate between locations. Despite its potential, scientific literature in multimodal networks has mostly focused on specific applications. The lack of a theoretical framework for the study of multimodal networks is evident. Applications of multimodal networks preferentially target freight and urban transportation systems. However, research in fields that might be regarded as far from logistics such as biology, computer networks and game theory have began to utilize multimodal networks as a modelling tool.

Traditionally, the modelling of multimodal networks has been primarily performed by adapting single–edge graphs where vertices represent physical locations such as hubs, warehouses and suchlike, whereas edges emulate roads, railways or motorway connections. The inclusion of transport modes in the model is not immediately perceived under this modelling approach. Similarly, the optimization of multimodal networks does not provide clarity about the treatment of the modes during analysis since they are mostly removed by reduction techniques and application–specific constraints in order to make the problem tractable.

This thesis introduced the coloured–edge graph which is a graph–based approach for the modelling and analysis of multimodal networks. The mainstay of this graph technique is simple: use a multigraph and label its edges with as many colours as transport mode there are, where each colour represents a specific transport choice for connecting two vertices. Unlike previous approaches, modes are explicitly represented in the graph and kept throughout analysis. A particular problem studied in this work has been the determination of minimal paths in the coloured–edge graph. Minimal paths are paths in the coloured–edge graph that are minimal in a sense of a partial order relation on paths weights.

Literature in the theoretical study of multimodal networks is scarce as the first part of this thesis has shown. Likewise, coloured-edge graphs were noted to be rarely investigated in the literature. Only two papers were identified for this research directly addressing the concept of coloured–edge graphs (see Chapter 2). Importantly, the most recent of these papers claims that optimization problems related to coloured–edge graphs such as the computation of the shortest paths still remains as a novel topic.

As a conclusion, the coloured–edge graph constitutes a powerful and straightforward modelling tool for the study of multimodal networks. However, more theoretical research into multimodal networks is still an issue that needs to be addressed. In addition, more study on application fields other than only freight and urban transportation is worth being researched.

7.2 Tractability of the Approach

This thesis addressed one particular problem associated with the coloured–edge graph. This is the computation of minimal paths. This problem states that given a weighted coloured–edge graph and a source vertex s, compute all the minimal paths from s that satisfy a product order partial relation defined on its path weights.

The computation of minimal paths is executed by a generalization of the well–know Dijkstra's algorithm. The implementation of the algorithm required the development of a partially ordered data structure. The imposition of an order relation on path weights does not produce a unique solution to this problem so that an optimal solution has to be obtained by selecting a path from the set of solutions according to some rule imposed by a potential user. Hereby, the multimodal Dijkstra's algorithm introduced in Chapter 3 produces a set of minimal paths. The cardinality of this set determines the tractability of the problem. The benefit in generating an entire set of optimal paths is twofold. First, it gives more decision power to a potential user since all minimal combinations of modes are listed. Second, post–optimal analysis can be performed without re–running the algorithm.

The number of minimal paths was firstly studied experimentally. Random coloured–edge graphs were used as inputs for the multimodal Dijkstra's algorithm. The results for complete random coloured–edge graphs with uniform weights were promising. The cardinality in this case became low order polynomial in most instances. At the other extreme pathological instances based on coloured–edge chains showed exponential orders in the number of minimal paths. Coloured–edge graphs with Euclidean weights were also studied. Remarkably good performances of the algorithm were obtained for higher dimensions. Data structures for handling weight tuples were also tested. To this end, the multimodal Dijkstra's algorithm was implemented with a priority queue, a simple list and a DAG data structure. Tests based on complete coloured–edge graphs with uniform weights revealed the priority queue as the most efficient data structure for handling minimal paths in a random coloured–edge graph.

The experiments with random coloured–edge graphs raised a series of question regarding the behaviour of the number of minimal paths in a best, average and worst case scenario. Chapters 4 and 5 sought answers for these question by means of the determination of upper and lower bounds on the number of minimal paths. It was discovered that the number of minimal paths is bounded above by k^{n-1} , where k and n stand for the number of colours and vertices, respectively. This bound is tight and demonstrates the potential intractability of the minimal path problem in coloured–edge graphs. This result is also consistent with the results encountered in the experiments with pathological instances. Specifically, coloured–edge chains whose weights follow very special patterns are likely to produce an exponential number of minimal paths. For instance, a high level of correlation among weights in a chain tend to result in more minimal paths. In

Chapter 4 the number of minimal paths was bounded below by k for complete coloured–edge graphs. This shows that there is a big range between the best case and the worst case cardinalities.

As a complement of the bounds, the $\mathcal{NP}$–completeness of the problem was shown. A reduction from the bin packing problem indicated the computation of minimal paths in a weighted coloured–edge graph has nondeterministic polynomial time.

Despite the potential exponential behaviour evidenced by the upper bound on the number of minimal paths, random instances provide more tractable cases. The experiments with random coloured–edge graphs seemed to support this idea. A semi–random input model was the main tool used in the development of probabilistic bounds for the coloured–edge graph. Although this model does not explain what happens in an average case, it does yield an understanding about the behaviour of the coloured–edge graph with probabilistic inputs. This approach turns out to be more realistic than a pure random setting because it is difficult to interpret in reality what a random input is. The analysis delivered an $O(n^3)$ bound for the expected number of minimal paths in bicoloured–edge graphs whose weights in one colour are randomly chosen.

As a conclusion, the computation of minimal paths in real multimodal networks is typically computationally tractable when the coloured–edge graph is used as modelling tool. Such tractability is attainable without resorting to reduction techniques, special constraints or heuristics. The semi–random input model also provides a method by which tractability can be "controlled" in a coloured–edge graph since intractable cases can be handled by slightly perturbing their edge weights. Additionally, the conditions that a multimodal network has to meet in order to produce an exponential number of paths are so specific that intuitively they are unlikely to occur in a random instance. This might be an indication that the number of minimal paths are predominantly concentrated between these two extremes.

Future work can investigate approximation algorithms based on semi–random instances of the coloured–edge graph. More theoretical as well as experimental studies in efficient data structures is also needed. For example, the outperformance of the QUEUE over the DAG requires more analysis.

Another source of research in coloured–edge graphs is the development of an average case for coloured–edge graphs with higher number of colours since it could clarify whether the polynomial behaviour is still obtainable. How concentrated the number of minimal paths are around its expected value is another issue that deserves investigation. As worst and best case require very special settings to occur, so minimal paths are mostly found between these two extremes. However, it is unknown whether general instances of the coloured–edge graph predominantly concentrate their minimal sets closer to a best or worst case. Finally, the probabilistic bounds for multicriteria optimization problems presented at the end of Chapter 5 might provide evidence that an improvement of the probabilistic bounds is attainable. However, it remains unclear whether the probabilistic input models utilized in these studies are replicable for instances of the coloured–edge graph.

7.3 Multicriteria Optimization Problems

This thesis also investigated the applicability of the coloured–edge graph in the modelling of multicriteria problems whose structure is combinatorial, specifically, the Multicriteria Shortest Path Problem.

A transformation based on the division of edges enables a Multicriteria Shortest Path Problem to be turned into shortest path problem in a coloured–edge graph. Such a transformation is algorithmically tractable. The use of the coloured–edge graph in the modelling of these problems permits the use of combinatorial tools such as counting arguments that might help in the development of more accurate bounds for multicriteria combinatorial problems.

An important contribution of this analysis was the determination of two cases in which all the paths in the graph were minimal. The construction considered biweighted complete graphs. These instances might question the exponential upper bound provided by Hansen for the bicriteria shortest path problem since the number of paths in a complete graph has factorial order. Future work can study the generalization of the constructions presented in the last section of Chapter 4.

7.4 Applicability of the Approach

Real multimodal networks were used in Chapter 6 of this thesis for evaluating the performance of coloured–edge graph as modelling tool. Three cases were analyzed.

The setup in Case (1) sought to study the computation of minimal paths in small instances. Multimodal networks from New Zealand and some European countries were the inputs. The number of vertices ranged between 120 and 900. The maximum number of colours (modes) were three. An important consideration in this case was the degree to which network shape influences the number of minimal paths. Countries with long shapes reported higher numbers of minimal paths. The resemblance of these long shapes to coloured–edge chains partly explains the large number of minimal paths. The other factor contributing is the trend in building mode networks very close to each other. For instance, railway systems tend to follow alongside to roadways systems in real systems. This creates a certain level of correlation between edge weights in different modes. As a result, weight patterns arise that might produce a high number of minimal paths.

Case (2) was set in order to determine the effect of n on the number of minimal paths and the time required to determine them. The setup was a large system of junctions that describes the transportation system of France. Four transport modes were used. By first applying a cluster routine to interconnect the different modes, the multimodal Dijkstra's algorithm computed minimal paths for several clustering radii for the France network. As far as this research is concerned, these instances are as large as those presented in literature about multimodal networks. More importantly, no restrictions or reductions whatsoever were needed to be applied to obtain the minimal paths.

Unlike Case (2), Case (3) focused more on the influence of k in the computation of minimal paths. Thus, the selected multimodal network (an international airline system) employed airlines as colours. This variable took values between 2 and 7 colours. The number of vertices was just 50, much smaller than the France multimodal network. Moreover, computations were performed by comparing two data structures: the priority queue and the DAG. The computations were longer as the number of airlines increased. A significant difference was noticed in the performance of the queue with respect to the DAG, the former by far being faster.

From Case (2) and (3) it is concluded that the number of colours (k) is a more limiting factor for the computations than the number of vertices in the network (n). The improvement of data structures requires more research.

Although a few of the running times obtained in the practical study could be regarded as excessive, it cannot be overlooked that these times can be considerably reduced by network reduction techniques. For instance, all intercity connections were considered as bidirectional, a fact that in reality is not true. Additionally, the topology of these real networks significantly differs from simple networks such as lattice or random graphs [1]. In other words, a more realistic modelling produces a series of constraints that might help the performance of the computations. The main purpose of the practical study was to push the multimodal Dijkstra's algorithm as far as possible in order to establish the degree of "computational amenability" of the coloured–edge graph.

7.5 Open Problems

The analysis of coloured–edge graphs is a nascent research field. Resulting from this thesis, a series of open problems involving coloured–edge graphs have arisen.

- The determination of exact asymptotics for the data structures presented in Chapter 3. The better performance obtained by the QUEUE over the DAG requires more theoretical analysis.
- To corroborate the existence of a factorial upper bound on the number of solutions for the bicriteria shortest path problem. The constructions presented at the end of Chapter 4 suggest the existence of an upper bound that might be of factorial order in n rather than exponential order.
- The determination of a bound on the expected number of minimal paths for coloured–edge graphs with $k > 2$. In other words, the determination of

[1] In network theory, the study of the topology of real networks is part of a branch know as"Complex Networks".

a bound on the average number of minimal paths in higher dimensions of the colour space.

- The distribution of the cardinality of $\mathcal{M}_{uv}$ around its mean. This can be done by establishing a concentration bound. In probability theory a concentration bound measures the deviation of a random variable from its expected value, Steele [87]. The determination of such a bound might significantly improve the probabilistic analysis of the coloured–edge graph since it can clarify whether a typical instance of the coloured–edge graph is closer to a worst or best case scenario.

- The observed reduction of the number of minimal paths for coloured–edge graphs with Euclidean weights in higher dimensional spaces requires more analysis. This work explained that a coloured–edge graph in a one–dimensional Euclidean space results in a worst case, so this case has to be avoided in order to obtain better performance. However, the nature of the reduction in higher dimensions remains unclear.

- Whether the the probabilistic bounds in Chapter 5 can be improved. This answer might be partly answered by Brunsch & Röglin [77]. However, they obtained probabilistic bounds for the multicriteria combinatorial optimization problem. Possibly the approach used by these authors might be able to be applied in the context of coloured–edge graphs.

Appendix A

France Input Dataset

The following data are a partial view of the input required by the multimodal Dijkstra's algorithm in order to compute minimal paths. For sake of simplicity, files of just one transport mode are depicted (motorways). Each transport mode has the same set of files. The abbreviations MW (motorways), PR (primary road), RW (railway), and WW (waterways) stand for the transport modes. The connectivity of the mode network is provided by the file "MWedges.txt". The rest of the text files represent interjunction distances which are measured as straight distance between a junction in a mode and the closest junction in another mode (e.g. "MWtoRW.txt"). These files are generated by using the ArcGIS tool "join and relates". The main purpose of these interjunction distances is the computation of a clustering that permits the generation of the final multimodal network.

```
------ File: MWedges.txt ------
#from to distance
0 2 0.0198899315658
1 3 0.0198604670494
2 0 0.0198899315658
2 4 0.000493028972852
3 1 0.0198604670494
3 5 0.000494609084695
5 3 0.000494609084695
5 7 0.000510371492751
⋮
```

```
------ File: MWtoRW.txt ------
#from, to, distance
0, 541, 0.019053
1, 541, 0.019168
2, 547, 0.024415
3, 547, 0.024539
4, 547, 0.024646
5, 547, 0.024769
6, 547, 0.024901
7, 547, 0.025013
⋮
```

```
------ File: MWtoPR.txt ------
#from, to, distance
0, 2618, 0.016378
1, 2621, 0.016326
2, 2625, 0.006981
3, 2625, 0.007105
4, 2625, 0.007275
5, 2625, 0.007397
6, 2625, 0.007611
7, 2625, 0.007719

⋮
```

```
------ File: MWtoWW.txt ------
#from, to, distance
0, 1015, 0.010086
1, 1015, 0.010190
2, 1015, 0.024010
3, 1015, 0.024113
4, 1015, 0.024458
5, 1015, 0.024562
6, 1015, 0.024942
7, 1015, 0.025026

⋮
```

As a reminder, the motorway system (MW) has 7720 junctions and 7432 polylines. The final multimodal network is comprised of four transport modes that together provide 97066 junctions and 86810 edges.

Appendix B

Airline Data

The following data are a partial view of the input required by the multimodal Dijkstra's algorithm in order to compute minimal paths for the global airline network. The multimodal Dijkstra's algorithm takes three text files as inputs in order to produce a set of minimal paths. The file "airlines.txt" is a top 10 ranking of international airlines based on number of passengers carried in decreasing order. The file "routes.txt" has over 60000 airport connections shown by airline. Finally, the file "airports.txt" contains a list with the 50 busiest airports in the world together with their corresponding longitude and latitude coordinates (in decimal degrees). All the presented information is coded according to IATA notation (www.iata.com).

```
------ File: airlines.txt ------   ------ File: routes.txt ------
# airline-id airline-name          # airline from to
DL Delta                           0B AGP BBU
UA United Airlines                 0B BBU AGP
WN Southwest Airlines              0B BBU BCN
AA American Airlines               0B BBU BGY
LH Lufthansa                       0B BBU BLQ
CZ China Southern AW               0B BBU BRU
FR Ryanair                         0B BBU BVA
AF Air France                      0B BBU CTA
KL KLM                             0B BBU DUB
MU China Eastern Airlines          0B BBU FCO
                                   ⋮
```

```
------ File: airports.txt ------
#IATA latitude longitude AIRPORT
ATL 33.63670 -84.42810 United States Hartsfield-Jackson Atlanta International Airport
LHR 51.47750 -0.46140 United Kingdom London Heathrow Airport
PEK 40.07330 116.59500 People's Republic of China Beijing Capital International Airport
ORD 41.97860 -87.90480 United States O'Hare International Airport
HND 35.55330 139.78120 Japan Tokyo International Airport
CDG 49.00970 2.54780 France Paris Charles de Gaulle Airport
LAX 33.94250 -118.40810 United States Los Angeles International Airport
DFW 32.89680 -97.03800 United States Dallas-Fort Worth International Airport
FRA 50.03330 8.57050 Germany Frankfurt Airport
DEN 39.86170 -104.67320 United States Denver International Airport
MAD 40.47220 -3.56090 Spain Madrid-Barajas Airport
JFK 40.63970 -73.77890 United States John F. Kennedy International Airport
HKG 22.30890 113.91460 Hong Kong Hong Kong International Airport
AMS 52.30810 4.76420 Netherlands Amsterdam Airport Schiphol
DXB 25.25280 55.36440 United Arab Emirates Dubai International Airport
BKK 13.68110 100.74730 Thailand Suvarnabhumi Airport
LAS 36.08040 -115.15230 United States McCarran International Airport
IAH 29.98440 -95.34140 United States George Bush Intercontinental Airport
PHX 33.43420 -112.01150 United States Phoenix Sky Harbor International Airport
SFO 37.61890 -122.37490 United States San Francisco International Airport
SIN 1.35920 103.98940 Singapore Singapore Changi Airport
CGK -6.12390 106.66110 Indonesia Soekarno-Hatta International Airport
CAN 23.39000 113.30670 People's Republic of China Guangzhou Baiyun International Airport
CLT 35.21400 -80.94310 United States Charlotte Douglas International Airport
MIA 25.79330 -80.29060 United States Miami International Airport
FCO 41.80030 12.23890 Italy Leonardo da Vinci Airport
MCO 28.42940 -81.30900 United States Orlando International Airport
SYD -33.94670 151.17670 Australia Kingsford Smith Airport
EWR 40.69250 -74.16870 United States Newark Liberty International Airport
MUC 48.35380 11.78610 Germany Munich Airport
LGW 51.14810 -0.19030 United Kingdom London Gatwick Airport
MSP 44.88200 -93.22180 United States Minneapolis-Saint Paul International Airport
NRT 35.76470 140.38640 Japan Narita International Airport
PVG 31.14170 121.79000 People's Republic of China Shanghai Pudong International Airport
DTW 42.21240 -83.35340 United States Detroit Metropolitan Wayne County Airport
SEA 47.44900 -122.30930 United States Seattle-Tacoma International Airport
PHL 39.87190 -75.24110 United States Philadelphia International Airport
YYZ 43.67720 -79.63060 Canada Toronto Pearson International Airport
IST 40.97610 28.81390 Turkey Atat??rk International Airport
KUL 2.74330 101.69810 Malaysia Kuala Lumpur International Airport
ICN 37.46250 126.43920 South Korea Seoul Incheon International Airport
BCN 41.29710 2.07850 Spain Barcelona Airport
BOS 42.36430 -71.00520 United States Logan International Airport
DEL 28.56870 77.11210 India Indira Gandhi International Airport
MEL -37.67330 144.84331 Australia Melbourne Airport
ORY 48.72330 2.37940 France Paris-Orly Airport
SHA 31.20000 121.33330 People's Republic of China Shanghai Hongqiao International Airport
BOM 19.09080 72.86670 India Chhatrapati Shivaji International Airport
SZX 22.63830 113.81170 People's Republic of China Shenzhen Bao'an International Airport
MEX 19.43630 -99.07220 Mexico Mexico City International Airport
```

References

[1] Heath, L., Sioson, A.: Semantics of multimodal network models. Computational Biology and Bioinformatics, IEEE/ACM Transactions on **6**(2) (2009) 271 –280

[2] Kilgour, D.M., Hipel, K.W.: Conflict analysis methods: The graph model for conflict resolution. In Kilgour, D.M., Eden, C., eds.: Handbook of Group Decision and Negotiation. Volume 4 of Advances in Group Decision and Negotiation. Springer Netherlands (2010) 203–222

[3] European-Parliament: Intermodality of goods transportation. Technical report, European Commission, Brussels (1997)

[4] Krebs, R.D.: Toward a national intermodal transportation system. Transportation Quarterly **48**(4) (1994) 333 – 342

[5] Ahuja, R.K., Magnanti, T.L., Orlin, J.B., Reddy, M.: Chapter 1 applications of network optimization. In M.O. Ball, T.L. Magnanti, C.M., Nemhauser, G., eds.: Network Models. Volume 7 of Handbooks in Operations Research and Management Science. Elsevier (1995) 1 – 83

[6] Abrach, H., Bhatti, S., Carlson, J., Dai, H., Rose, J., Sheth, A., et al.: Mantis: system support for multimodal networks of in-situ sensors. In: Proceedings of the 2nd ACM international Conference on Wireless Sensor Networks and Applications, San Diego, CA. (2003) 50–59

[7] Chen, J., Dougherty, E, Demir, S, S., Friedman, C: Grand challenges for multimodal bio-medical systems. In: Circuits and Systems Magazine, IEEE (2005) 46 – 52

[8] Heath, L., Sioson, A.: Multimodal networks: Structure and operations. IEEE/ACM Transactions on Computational Biology and Bioinformatics **99**(1) (2007) 1–19

[9] Sioson, A.A.: Multimodal Networks in Biology. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia (2005)

[10] Medeiros, D.J., Traband, M., Tribble, A., Lepro, R., Fast, K., Williams, D.: Simulation based design for a shipyard manufacturing process. In: Simulation Conference Proceedings. (2000) 1411 – 1414

[11] Kiesmüller, G.P., de Kok, A.G., Fransoo, J.C.: Transportation mode selection with positive manufacturing lead time. Transportation Research Part E **41** (2005) 511 – 530

[12] Nigay, L., Coutaz, J.: A design space for multimodal systems: concurrent processing and data fusion. In: CHI '93: Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems, New York, NY, USA, ACM (1993) 172–178

[13] Boyce, D.: Forecasting travel on congested urban transportation networks:review and prospects for network equilibrium models. Networks and Spatial Economics **7**(2) (2007) 99 – 128

[14] Macharis, C., Bontekoning, Y.M.: Opportunities for or in intermodal freight transport research: A review. European Journal of Operational Research **153**(2) (2004) 400 – 416 Management of the Future MCDA: Dynamic and Ethical Contributions.

[15] Hillier, F., Lieberman, G.: Introduction to Operation Research. McGraw-Hill (2005)

[16] Min, H.: International intermodal choices via chance-constrained goal programming. Transportation Research Part A: General **25**(6) (1991) 351 – 362

[17] Kim, D., Barnhart, C., Ware, K., Reinhardt, G.: Multimodal express package delivery: A service network design application. Transportation Science **33**(4) (1999) p391 –

[18] Kim, B.J., Kim, W.: An equilibrium network design model with a social cost function for multimodal networks. Annals of Regional Science **40**(3) (2006) 473 – 491

[19] Horn, M.E.T.: An extended model and procedural framework for planning multi-modal passenger journeys. Transportation Research Part B: Methodological **37**(7) (2003) 641 – 660

[20] Chang, T.S.: Best routes selection in international intermodal networks. Computers and Operations Research **1** (2007) 1 – 15

[21] Jarzemskiene, I.: The evolution of intermodal transport research and its development issues. TRANSPORT **22**(4) (2007) 296 – 306

[22] Nagurney, A.B.: Comparative test of multimodal traffic equilibrium methods. Transportation Research B **18**(6) (1984) 469 – 485

[23] Ayed, H., Khadraoui, D., Habbas, Z., Bouvry, P., Merche, J.F.: Transfer graph approach for multimodal transport problems. In: Modelling, Computation and Optimization in Information Systems and Management Sciences. (2008) 538–547

[24] Foo, H.M., Leong, H.W., Lao, Y., Lau, H.C.: A multi-criteria, multi-modal passenger route advisory system. In: Proc. 1999 IES-CTR Int'l Symp., Singapore (1999)

[25] Qiang Li, C.E.K.: Gis-based itinerary planning system for multimodal and fixed-route transit network. In: Proceedings of the Mid-Continent Transportation Symposium 2000. (2000)

[26] Kitamura, R., Chen, N., Chen, J.: Daily activity and multimodal travel planner final report. Technical report, UCB-ITS-PRR-99-1 (1999)

[27] Fragouli, M., Delis, A.: Easytransport: an effective navigation and transportation guide for wide geographic areas. In: Tools with Artificial Intelligence, 2002. (ICTAI 2002). Proceedings. 14th IEEE International Conference on. (2002) 107–113

[28] Lawler, E.: Combinatorial Optimization, Networks and Matroids. Dover Publication, INC, New York (2001)

[29] Lozano, A., Storchi, G.: Shortest viable hyperpath in multimodal networks. Transportation Research Part B: Methodological **36**(10) (2002) 853 – 874

[30] Tarapata, Z.: Selected multicriteria shortest path problems: An analysis of complexity, models and adaptation of standard algorithms. International Journal of Applied Mathematics and Computer Science **17** (2007) 269–287

[31] Soroush, H.: Optimal paths in bi-attribute networks with fractional cost functions. European Journal of Operational Research **190**(3) (2008) 633 – 658

[32] Androutsopoulos, K.N., Zografos, K.G.: Solving the multi-criteria time-dependent routing and scheduling problem in a multimodal fixed scheduled network. European Journal of Operational Research **192**(1) (2009) 18 – 28

[33] Aifadopoulou, G., Ziliaskopoulos, A., Chrisohoou, E.: Multiobjective optimum path algorithm for passenger pretrip planning in multimodal transportation networks. Transportation Research Record **2032** (2007) 26–34

[34] Modesti, P., Sciomachen, A.: A utility measure for finding multiobjective shortest paths in urban multimodal transportation networks. European Journal of Operational Research **111**(3) (1998) 495 – 508

[35] Hansen, P.: Bicriterion path problems. In: G. Fandel and T. Gal, Editors, Multiple Criteria Decision Making: Theory and Applications, Springer-Verlag, Heidelberg (1980)

[36] Loui, R.P.: Optimal paths in graphs with stochastic or multidimensional weights. Commun. ACM **26**(9) (1983) 670–676

[37] Müller-Hannemann, M., Weihe, K.: On the cardinality of the pareto set in bicriteria shortest path problems. Annals of Operations Research **147**(1) (2006) p269 – 286

[38] Röglin, H., Vöcking, B.: Smoothed analysis of integer programming. In: In Proceedings of the 11th International Conference on Integer Programming and Combinatorial Optimization (IPCO, Springer (2005) 276–290

[39] Lillo, F., Schmidt, F.: Optimal paths in real multimodal transportation networks: An appraisal using gis data from new zealand and europe. In: Operation Research Society of New Zealand: 45th Annual Conference. (2010) 281–288

[40] Ensor, A., Lillo, F.: Partial order approach to compute shortest paths in multimodal networks. Transportation Research C: Emerging Technologies. Under review (2011)

[41] Ensor, A., Lillo, F.: Partial order approach to compute shortest paths in multimodal networks. In: 23rd European Conference on Operational Research. (2009)

[42] Ensor, A., Lillo, F.: Tight upper bound on the number of optimal paths in weighted coloured-edge graphs. In: 24th European Conference on Operations Research. (July 2010)

[43] Ensor, A., Lillo, F.: Analyzing optimal paths in coloured–edge graphs with euclidean weights. In: 24rd European Conference on Operational Research. (2010)

[44] Malaguti, E., Toth, P.: A survey on vertex coloring problems. International Transactions in Operational Research **17** (2010) 1–34

[45] Du, D., Pardalos, P.M. In: Handbook of combinatorial optimization. Volume 2. Kluwer Academic Publishers (1998) 331–395

[46] Clímaco, J., Captivo, M., Pascoal, M.: On the bicriterion - minimal cost/minimal label - spanning tree problem. European Journal of Operational Research **204**(2) (2010) 199 – 205

[47] Xu, H., Li, K.W., Kilgour, D.M., Hipel, K.W.: A matrix-based approach to searching colored paths in a weighted colored multidigraph. Applied Mathematics and Computation **215**(1) (2009) 353 – 366

[48] Manoussakis, Y.: Alternating paths in edge-colored complete graphs. Discrete Applied Mathematics **56**(2-3) (1995) 297 – 309 Fifth Franco-Japanese Days.

[49] Martins, E.Q.V.: On a multicriteria shortest path problem. European Journal of Operational Research **16**(2) (May 1984) 236–245

[50] Ehrgott, M.: Multicriteria Optimization. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005)

[51] Ehrgott, M., Gandibleux, X.: A survey and annotated bibliography of multiobjective combinatorial optimization. OR Spectrum **22** (2000) 425–460

[52] Gross, J., Yellen, J.: Graph Theory and Its Applications. Second edn. Discrete Mathematics and Its Applications. CRC Press Series (1999)

[53] Raith, A., Ehrgott, M.: A comparison of solution strategies for biobjective shortest path problems. Computers & Operations Research **36**(4) (2009) 1299 – 1331

[54] Martello, S., Toth, P.: Knapsack problems: algorithms and computer implementations. John Wiley & Sons, Inc., New York, NY, USA (1990)

[55] Tung, C.T., Chew, K.L.: A multicriteria pareto-optimal path algorithm. European Journal of Operational Research **62**(2) (1992) 203 – 209

[56] Guerriero, F., Musmanno, R.: Label correcting methods to solve multicriteria shortest path problems. Journal of Optimization Theory and Applications **111** (2001) 589–613

[57] Ehrgott, M., Gandibleux, X.: Bound sets for biobjective combinatorial optimization problems. Computers & Operations Research **34**(9) (2007) 2674 – 2694

[58] Ehrgott, M., Gandibleux, X.: Approximative solution methods for multiobjective combinatorial optimization. TOP **12** (2004) 1–63 10.1007/BF02578918.

[59] Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik **1** (1959) 269–271

[60] Sniedovich, M.: Dijkstra's algorithm revisited: the dynamic programming connexion. Control and Cybernetics **35** (2006) 599–650

[61] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, Second Edition. The MIT Press (September 2001)

[62] Lozano, A., Storchi, G.: Shortest viable path algorithm in multimodal networks. Transportation Research Part A: Policy and Practice **35**(3) (2001) 225 – 241

[63] Bentley, J.L., Kung, H.T., Schkolnick, M., Thompson, C.D.: On the average number of maxima in a set of vectors and applications. J. ACM **25**(4) (1978) 536–543

[64] Müller-Hannemann, M., Weihe, K.: Pareto shortest paths is often feasible in practice. In: WAE '01: Proceedings of the 5th International Workshop on Algorithm Engineering, London, UK, Springer-Verlag (2001) 185–198

[65] Daskalakis, C., Karp, R.M., Mossel, E., Riesenfeld, S., Verbin, E.: Sorting and selection in posets. In: Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms. SODA '09, Philadelphia, PA, USA, Society for Industrial and Applied Mathematics (2009) 392–401

[66] Graham, R.L., Knuth, D.E., Patashnik, O.: Concrete Mathematics: A Foundation for Computer Science. 2nd edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1994)

[67] Serafini, P.: Some considerations about computational complexity for multi objective combinatorial problems. In: Recent advances and historical development of vector optimization. Volume 294 of Lecture Notes in Economics and Mathematical Systems. Springer Verlag, Berlin (1986) 222–232

[68] Ross, S.M.: Probability models for computer science. San Diego :Harcourt Academic Press (2002)

[69] Bramel, J., Simchi-Levi, D.: The logic of logistics : theory, algorithms, and applications for logistics management. Springer, New York (1997)

[70] Spielman, D.A., Teng, S.H.: Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. J. ACM **51**(3) (2004) 385–463

[71] Manthey, B., Reischuk, R.: Smoothed analysis of binary search trees. In: Proc. of the 16th Annual International Symposium on Algorithms and Computation (ISAAC. (2005) 483–492

[72] Spielman, D.A., ., S.H.T.: Smoothed analysis: Motivation and discrete models. In: Proceedings of WADS: 8th Workshop on Algorithm and Data Structures. (2003)

[73] Beier, R., Vöcking, B.: Random knapsack in expected polynomial time. J. Comput. Syst. Sci. **69** (November 2004) 306–329

[74] Beier, R., Röglin, H., Vöcking, B.: The smoothed number of pareto optimal solutions in bicriteria integer optimization. In Fischetti, M., Williamson, D., eds.: Integer Programming and Combinatorial Optimization. Volume 4513 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2007) 53–67

[75] Röglin, H., Teng, S.H.: Smoothed analysis of multiobjective optimizaton. In: Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Sciences. (2009) 681–690

[76] Moitra, A., O'Donnell, R.: Pareto optimal solutions for smoothed analysis. Technical report, CoRR(abs/1011.2249) (2010)

[77] Brunsch, T., Röglin, H.: Lower bounds for the smoothed number of pareto optimal solutions. Technical report, University of Bonn (2010)

[78] Zhan, F., Noon, C.: Shortest path algorithms: An evaluation using real road networks. Transportation Sciences **32**(1) (1998) 65–73

[79] Ayed, H., Galvez-Fernandez, C., Habbas, Z., Khadraoui, D.: Solving time-dependent multimodal transport problems using a transfer graph model. Computers & Industrial Engineering **In Press, Corrected Proof** (2010)

[80] Von Ferber, C., Holovatch, T., Holovatch, Y., Palchykov, V.: Public transport networks: empirical analysis and modeling. Eur. Phys. J. B **68**(2) (2009) 261–275

[81] Geofabrik: Europe shapefiles (http://www.geofabrik.de/ 2010)

[82] OpenFlights: Airport data (http://www.openflights.org/data.html 2010)

[83] Burke, R.: Getting to Know Arcobjects. ESRI Press (2002)

[84] Bielli, M., Boulmakoul, A., Mouncif, H.: Object modeling and path computation for multimodal travel systems. European Journal of Operational Research **175**(3) (2006) 1705 – 1730

[85] Ziliaskopoulos, A., Wardell, W.: An intermodal optimum path algorithm for multimodal networks with dynamic arc travel times and switching delays. European Journal of Operational Research **125**(3) (2000) 486 – 502

[86] Caramia, M., Guerriero, F.: A heuristic approach to long-haul freight transportation with multiple objective functions. Omega **37**(3) (2009) 600 – 614

[87] Steele, J.M.: Probability Theory and Combinatorial Optimization. SIAM (1997)