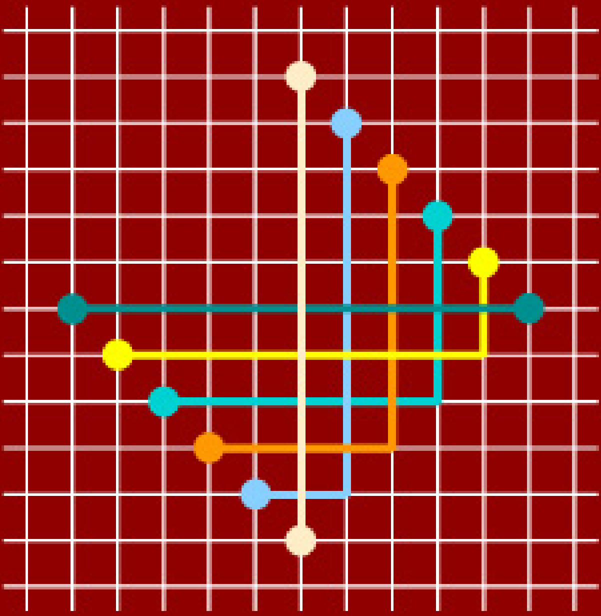
Ines Spenke

Complexity and Approximation of Static k–Splittable Flows and Dynamic Grid Flows



Cuvillier Verlag Göttingen

Complexity and approximation of static k–splittable flows and dynamic grid flows

vorgelegt von
Dipl.-Math. oec. Ines Spenke

Von der Fakultät II – Mathematik und Naturwissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
– Dr. rer. nat. –

genehmigte Dissertation

Berichter: Prof. Dr. Rolf H. Möhring
Prof. Dr. Ekkehard G. Köhler

Tag der wissenschaftlichen Aussprache: 21. Dezember 2006

Berlin 2006
D 83

Bibliografische Information Der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.ddb.de abrufbar.
1. Aufl. - Göttingen : Cuvillier, 2007
Zugl.: (TU) Berlin, Univ., Diss., 2006

978-3-86727-219-3

Nonnenstieg 8, 37075 Göttingen
Telefon: 0551-54724-0
Telefax: 0551-54724-21
www.cuvillier.de

1. Auflage, 2007
Gedruckt auf säurefreiem Papier

978-3-86727-219-3

Für Julia Elise Spenke.

Preface

The topic of *routing through a network* has attracted considerable attention from both the combinatorial optimization community and practitioners due to a variety of applications in logistics and traffic control. In combinatorial optimization, such routing questions are typically modeled as network flow problems. Obviously, these models cannot reflect all practical requirements. In this work, we consider two aspects that cannot be dealt with at all, or at least not efficiently with standard techniques.

Efficient models define network flows as properties associated with nodes and edges. In doing so, we have no control over the number of paths in a solution, although this is a key value for the solution's applicability. In the first part of this thesis, we consider so–called *k–splittable flows*. These flows are required to be splittable into no more than a given number k of paths. In correspondence with the classical maximum s,t–flow problem, we investigate the problem of finding a maximum k–splittable s,t–flow in a network with edge capacities. We study the complexity of this question for given integer values of k and for k that is a function on graph parameters. We describe solution methods for polynomially solvable variants and prove approximation results for NP–hard problem configurations. For the class of graphs with bounded treewidth, a new approach enables us to solve problem variants that are hard on general graphs.

In the second part, we add a time dimension to classical static flows. Flow can then be seen as moving over time through a network, which is essential for practical purposes. We study such *flows over time* on grid graphs that typically arise in storage facilities. We investigate the problem of sending a given amount of flow to targets within a minimum possible time. For this so–called *quickest flow problem*, we prove complexity results and show polynomial algorithms or approximations. An entire landscape of problem variants is encompassed: We consider single and multicommodity environments. Path flows can wait on edges or have to be sent without disruption. A temporarily closure of edges can be allowed. Transit times along edges are considered to be constant or edge–specific.

This thesis results from my work in the group “Combinatorial Optimization and Graph Algorithms” at the Technical University of Berlin. I have been assisted and supported by many people to whom I wish to express my thanks.

First of all, I want to thank Professor Rolf H. Möhring, who gave me the opportunity to work on flow problems in his group. I am especially grateful to him for raising my interest in the exciting topic of flows over time. This thesis would not exist without his helpful discussions and valuable hints.

I am very grateful to Ekkehard Köhler for introducing me to flows over time, for fruitful discussions on dynamic flows in grids, for useful ideas for proving NP–hardness and for taking the second assessment of this work.

In particular, I thank Martin Skutella whose flow knowledge I benefitted from substantially. The second chapter of this thesis is based on joint work with him and Ronald Koch, for which I cordially thank them both.

Many thanks to my COGA colleagues for numerous discussions on a variety of topics and for the pleasant atmosphere on our hall. Special thanks go to Sebastian Orlowski, Marco Lübbecke, Björn Stenzel and Sebastian Stiller for proofreading and for their helpful discussions on the topic.

This work was funded by the DFG “Mathematics for key technologies” research center, which provides excellent working conditions. My thanks go to all those who made Matheon possible.

My special thanks go to Mathias Schulz whose careful reading of this thesis and continuous encouragement enabled me to complete this work at just the right moment in life. Last but not least, I am very grateful to my family for supporting my interest in mathematics and for simply believing in me.

Berlin, February 2007 Ines Spenke

Contents

Chapter 1

Introduction

The questions studied in this thesis are based on routing problems at a container terminal at the Hamburg harbor. Between the harbor's quay and a storage area, more than sixty vehicles transport containers with a planned throughput of nearly two million containers per year. These vehicles are automatically guided to drive along specified lanes on their own. In order to transport all containers to their destinations, these vehicles have to be supplied with appropriate routes along the lanes to ensure that they do not collide with each other. Herein, time plays an essential role. Ships have to be loaded and discharged in a short time in order for the containers to be quickly brought to their destinations.

Such routing problems are a common task in logistics. Typically the aim is to find paths and to send transportation vehicles along them, such that given requests are fulfilled as quickly as possible with a limited number of vehicles. Most routing problems are complex due to a large amount of transportation requests and a large variety of possible routes in the given area. Therefore, good routings can hardly be determined manually, but a good control is a substantial competitive edge. As such, more and more sophisticated tools are currently under development to support this type of planning.

The field of combinatorial optimization provides methods to model and solve routing problems. Such mathematical methods are characterized not only by their ability to suggest strategies for a routing but also by their capability to prove the quality of the proposals, which other approaches normally cannot do. A broadly–applied method to model and solve routing problems within a combinatorial optimization framework are so–called network flows. Commonly, lanes are taken as edges and crossings as nodes of the network. Restrictions such as driving times, lane capacities, and costs for using a lane are assigned as parameters to edges and nodes of this routing network.

Every routing problem stemming from an application involves a number of special requirements based on the conditions of the processes to be modeled. Some of these restrictions are covered by standard network flow models, such as lane capacities and directions in which lanes can be traversed. Standard methods do not include a time dimension, although it is essential for routing strategies. As a consequence, waiting policies cannot be handled. Mostly, standard approaches allow fractional flow portions, meaning that path flows cannot be seen as vehicles. Bounds on the number of paths used cannot be expressed because when formulating flow edgewise concrete paths are ignored.

In this thesis, we investigate two extensions of the standard model. The first extension is motivated by the fact that transportation tasks often have to be fulfilled with a limited number of vehicles. Due to this number, only a bounded number of paths can be used. Standard network flow algorithms are not designed to respect such a bound, meaning that flows may travel along a huge number of paths. We take this bound into account and study so–called *k–splittable flows* using no more than k paths. We investigate a problem related to the well–known maximum s,t–flow problem by looking for *maximum k–splittable s,t–flows* where k is a given integer or depends on graph parameters.

In the second part, we involve a time dimension that cannot be handled efficiently by standard methods. We study so–called *flows over time* that allow the modeling of movements of flow through a network. We investigate such flows over time in networks featuring a special structure, namely grids that often occur in storage areas. Regarding these grid flows, we consider further aspects such as waiting on edges and time windows that close edges for certain time intervals. Such aspects motivated by practical requirements lead to interesting research questions.

For both k–splittable flows and flows over time in grids, we concentrate on analyzing the complexity and approximability of such problems. We identify polynomially solvable and NP–hard cases for a variety of problem variants. Non–approximability results as well as approximation algorithms are shown.

Both flow problems are related to the *disjoint paths problem*: For a given network with sources and sinks, the question is, how many source–sink pairs can be connected by pairwise edge–disjoint paths. Finding a maximum integral multicommodity k–splittable flow with unit edge capacities and a single path for each commodity corresponds to the same question. Maximum flows over time in graphs with edge capacities 1 and integral flow portions reflect the disjoint paths problem in a dynamic environment, such that we call it the *disjoint paths over time* problem. As far as we know, disjoint paths over time have not been studied at all up until now.

Outline of this thesis

The main part of this thesis consists of three chapters.

In Chapter 2, we investigate k–splittable flows. This chapter is based on joint work with Ronald Koch and Martin Skutella of the University of Dortmund. Parts of it are published in [30], [31], and [32]. It deals with the *maximum k–splittable s,t–flow problem* (MkSF), expanding on the well-known maximum s,t–flow problem with the requirement that the solution can be split into no more than k path flows. In this chapter, the value k is either a constant integer or is dependant on graph parameters. For both instances we study the complexity and approximability of the problem MkSF. An overview of polynomially solvable cases and NP–hard variants is developed and bounds for approximation guarantees are proven. For solving MkSF, a new approach is introduced by decomposing the problem into two subproblems, a packing and a routing step, which can be solved consecutively. We succeed in the unusual way of first determining path flow values and then looking for corresponding routes. With this strategy, we are able to calculate solutions for a special graph class, namely graphs of bounded treewidth: In the case of a constant number k, the problem is solved to optimality, whereas if k is part of the input, a polynomial time approximation scheme (PTAS) is derived.

We omit restrictions on the number of paths used and include a time dimension beginning with Chapter 3. Flows over time are considered in grid graphs. In Chapter 3, the single commodity case is investigated. We are interested in finding *quickest flows*, which means to determine the minimum possible time horizons needed to satisfy given demands. We consider a variety of specific problem configurations that differ by having uniform or edge–specific transit times. Secondly, waiting on edges can or cannot be allowed. Furthermore, edges can or cannot be closed for certain time intervals. In the case where edges are not temporarily closed, we provide polynomial solution methods. When in contrast to this, time windows close edges for certain periods of time, then some configurations are shown to be NP–hard. In some cases, the complexity remains open. Moreover, we prove certain non-approximability results.

In Chapter 4, the previous chapter is extended to more than one commodity. Again, the above–mentioned problem configurations are investigated. We show how to replace time windows with additional commodities. Thus, in contrast to the previous chapter, time windows are not the critical factor for the problem's complexity. All considered problem variants are proven to be NP–hard by a reduction from 3-COLORING. A second, much easier reduction from PARTITION, which applies only when non–uniform transit times are allowed, refines the results for some variants by already showing NP–hardness for very small constant numbers of commodities. We discuss the influence of waiting and of integrality requirements. Two approximation algorithms are introduced, one for all variants without time windows and the other for cases when additional transit times are uniform along all grid edges.

Preliminaries

We assume the reader is familiar with basic concepts of complexity theory. A good survey is provided by Garey and Johnson [19]. Some terms used in this thesis are briefly summarized in the following. Throughout this work we assume that $P \neq NP$.

A problem is said to be *strongly NP-hard* if it is still NP-hard even when the absolute values of all numbers in the input are bounded by some polynomial in the length of the input. Thus, NP-hard problems without numbers are always strongly NP-hard. An algorithm is called *pseudo–polynomial* if it is polynomial in its number of input values and in its maximum input value. If there is a pseudo–polynomial algorithm, then a problem is not strongly NP-hard. A problem that is NP-hard but not in the strong sense is called *weakly NP-hard*.

In the following chapters, we refer to some well–known NP–complete problems. To simplify the notation and to avoid redundancies, we outline them here and use the same notation throughout this work. More details can be found in Garey and Johnson's work [19].

- SUBSETSUM: Given q positive integers $u_1, ..., u_q$ and a number M, is there a subset $S \subseteq \{1, ..., q\}$, such that $\sum_{i \in S} u_i = M$?

- SAT: A set of variables $\{x_1, ..., x_r\}$ and a set of clauses $\{C_1, ..., C_q\}$ over the set of variables is given. Is there a truth assignment on the variable set satisfying all clauses?

- 3SAT: A set of variables $\{x_1, ..., x_r\}$ and a set of clauses $\{C_1, ..., C_q\}$ over the set of variables is given, such that each clause contains three variables. Is there a truth assignment for the variable set satisfying all clauses? This problem is NP–complete in the strong sense.

- Partition: Given positive integers $a_1, ..., a_r$, is there a partition of these integers into two groups with the same sum of elements?

- 3-Coloring: Is it possible to color the nodes of a given graph with three colors, such that adjacent nodes have different colors? The problem is NP–complete in the strong sense.

Some NP–hard problems can be handled by so–called *approximation algorithms*. These are polynomial algorithms that provide a feasible solution to a problem with a certain performance guarantee. Such guarantees are given as relative or absolute values referring to an optimal value OPT. An algorithm A is said to have a *relative guarantee* of α if, for all problem instances I, it gives a solution $A(I)$ with $A(I) \geq \alpha\, OPT(I)$ for a maximization problem and $A(I) \leq \alpha\, OPT(I)$ for a minimization problem. Obviously, $\alpha \leq 1$ in the case of a maximization problem and $\alpha \geq 1$ in a minimization problem. A has an *absolute guarantee* of β if its solutions fulfill $A(I) \geq OPT(I) + \beta$ in a maximization problem, and otherwise $A(I) \leq OPT(I) + \beta$. In a maximization problem $\beta \leq 0$ is required and in a minimization problem $\beta \geq 0$.

As a reference for the reader, a list of symbols is given at the end of this thesis.