Selfish versus Coordinated Routing in Network Games

by

Nicolás E. Stier-Moses

B.S., Universidad de Buenos Aires (1999)

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2004

Signature of Author . . . .
Sloan School of Management
May 14, 2004

Certified by . . . .
Andreas S. Schulz
Class of 1958 Associate Professor of Operations Research
Thesis Supervisor

Accepted by . . . .
John N. Tsitsiklis
Professor of Electrical Engineering and Computer Science
Co-director, Operations Research Center

Selfish versus Coordinated Routing in Network Games

by

Nicolás E. Stier-Moses

Submitted to the Sloan School of Management
on May 14, 2004, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Operations Research

Abstract

A common assumption in network optimization models is that a central authority controls the whole system. However, in some applications there are independent users, and assuming that they will follow directions given by an authority is not realistic. Individuals will only accept directives if they are in their own interest or if there are incentives that encourage them to do so. Actually, it would be much easier to let users make their own decisions hoping that the outcome will be close to the authority's goals. Our main contribution is to show that, in static networks subject to congestion, users' selfish decisions drive the system close to optimality with respect to various common objectives. This connection to individual decision making proves fruitful; not only does it provide us with insights and additional understanding of network problems, but it also allows us to design approximation algorithms for computationally difficult problems.

More specifically, the conflicting objectives of the users prompt the definition of a network game in which they minimize their own latencies. We show that the so-called price of anarchy is small in a quite general setting. Namely, for networks with side constraints and non-convex, non-differentiable, and even discontinuous latency functions, we show that although an arbitrary equilibrium need not be efficient, the total latency of the best equilibrium is close to that of an optimal solution. In addition, when the measure of the solution quality is the maximum latency, equilibria in networks without constraints are also near-optimal. We provide the first analysis of the problem of minimizing that objective in static networks with congestion. As this problem is NP-hard, computing an equilibrium represents a constant-factor approximation algorithm.

In some situations, the network authority might still want to do better than in equilibrium. We propose to use a solution that minimizes the total latency, subject to constraints designed to improve the solution's fairness. For several real-world instances, we compute traffic assignments of notably smaller total latency than an equilibrium, yet of similar fairness. Furthermore, we provide theoretical results that explain the conclusions derived from the computational study.

Thesis Supervisor: Andreas S. Schulz
Title: Class of 1958 Associate Professor of Operations Research

To Nati

Biographical Note

Nicolás Stier-Moses received a degree of *Licenciado en Ciencias Informáticas* from *Universidad de Buenos Aires* in April 1998. (This degree is equivalent to a combined Bachelor and Master's degree in Computer Science.) Afterwards, he spent one year as a research assistant at INRIA in Rennes, France. In 1999, he moved to the U.S. to join the Operations Research Center at MIT from which he will receive a Ph.D. degree in June 2004.

Acknowledgments

There were many who contributed and made the existence of this dissertation possible. First and foremost, I want to thank Andreas Schulz, my advisor, for always offering invaluable help and encouragement. His academic (and sometimes non-academic) advice, knowledge, and support were key in achieving my goals. Working with my other co-authors, Rolf Möhring and José Correa, was not only inspiring, it was a pleasure. I have greatly benefited from all the discussions that we carried over on the ORC whiteboards and elsewhere (i.e., ABP). I wish to thank Tom Magnanti and John Tsitsiklis, members of my Doctoral Committee, for their enlightening suggestions, questions, and insights along the way. I am indebted to Patricio Méndez, Carol Meyers and Fernando Ordoñez for their careful reading of preliminary versions of this dissertation and for numerous suggestions for improvement.

During these years, there were many others who helped in different ways. Thanks to Cindy Barnhart, Jérémie Gallien, Michel Goemans, Ramesh Johari, Jim Orlin, Asu Ozdaglar, and Tim Roughgarden for many interesting and stimulating discussions. The ORC students in general, and particularly Hernán Alperín, René Caldentey, Ozie Ergun, Samuel Fiorini, Paulo Gonçalvez, Martin Haugh, Nicole Linneberg, Ariel Schilkrut, and Mike Wagner made this period in Boston one of the greatest experiences of all my life. Thanks are also due to my friends from Argentina, who particularly helped me right after I moved to Boston. In addition, I want to acknowledge the support given by the *Projet Mascotte*; especially, I thank Afonso Ferreira for an excellent summer at INRIA. I am also grateful for the support offered by the ORC staff: Veronica Mignot, Paulette Mosley, and Laura Rose.

I want to mention a few others who greatly contributed to my education and offered their friendship. Leonardo Salama taught me the basics of computer science and remains one of the best professors I have ever had. Alfredo Vega Weiss and Graciela Deferrari transmitted to me the enthusiasm of doing research and solving problems. Isabel Méndez and Gerardo Rubino, the advisors I had in Argentina and France, taught me innumerable things and supported me while doing my first 'real' research project. From Jane Dunphy,

I learned a lot about teaching, but, in addition, I appreciated that she was there at a time when I needed some help. I have enjoyed the friendship of Ariel, Fer, José, and Pato, whom I especially thank.

Finally, I want to thank my family in Boston and in Argentina for being so accommodating to long working hours and to the time away from them. The love, support and encouragement that I received from all of them allowed me to go on even during difficult times. In particular, I thank my parents, their significant others, my in-laws, and their significant others. Last, but not least, my love goes to my wife Natalia, my daughter Sophie, and my son Martín.

Cambridge, May 2004 *Nicolás Stier-Moses*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

A common assumption in network optimization models is that a central authority controls the whole system, dictating what individuals should do. However, this assumption is not necessarily valid in many real-world applications. It is sometimes not realistic to assume that users of the system will follow directions given by the central authority, unless the directives are in their own interest or there are incentives that encourage them to do so. It would be much easier—and nicer—to let users make their own decisions, hoping that the outcome will be close to the central authority's objectives. For that reason, our goal is to explore systems subject to congestion from the perspective of the individuals. We will see that, in static network flow models, users' independent and selfish decisions inadvertently drive the solution towards optimality with respect to various objectives common in applications. This connection to individual decision making proves fruitful; not only does it provide us with insights and additional understanding of the problems we study, but it also allows us to design approximation algorithms for NP-hard problems.

Consider a network of arcs that will be used as a medium to route demands given by a collection of origin-destination pairs. To model congestion, we associate a *latency function* to every arc. Each of these functions maps the flow rate on the arc to the time that each user needs to traverse it. The existence of congestion generates interdependencies between users' decisions, which can be modeled using game theory. In our model, we assume that there are infinitely many users who are independent and select routes selfishly. We can expect to see a traffic pattern in which all users are taking their respective shortest paths under the prevailing congestion condition; otherwise some user would switch to a shorter route. Therefore, this solution is *fair*: different users traveling between the same terminals experience the same travel time. Moreover, such a flow is a *Nash equilibrium* of the traffic game we just introduced. We will use the standard terminology in traffic networks and call it a *user equilibrium*.

The network manager presumably cares about the system as a whole, and an objective that does not give preference to particular users is the total (or equivalently, average) travel time in the system. If the solution that minimizes this objective—typically referred to as a *system optimum*—is computed, users could be guided accordingly. However, unless they are forced, users will not have an incentive to follow those directives because it is well-known that some will end up making long detours. Namely, to reach a solution of minimal travel time some users need to sacrifice themselves for the community's benefit. This is not realistic and probably very few users would be willing to do so. For this reason, a result previously proved by Roughgarden and Tardos (2002) and Roughgarden (2003b) is very interesting. The total travel time of a user equilibrium is not higher than a small constant times that of a system optimum. For example, if the latency functions are linear, the total travel time of a user equilibrium is at most 33% higher than in a system optimum. Papadimitriou (2001) coined the term *price of anarchy* to refer to that multiplicative constant. The reason for using that term should be clear: It measures the cost of lacking central coordination. In other words, it represents how much the system loses by letting users make their own choices. This concept, first proposed by Koutsoupias and Papadimitriou (1999) for a particular telecommunication network model, has recently received considerable attention. Although this dissertation considers network games, these ideas can be used to characterize the efficiency of Nash equilibria in other applications as well, as we shall discuss in Chapter 7.

One of our main contributions is an extension of the results of Roughgarden and Tardos to networks with *side constraints* and to latency functions that are *non-convex*, *non-differentiable*, and even *discontinuous*. Side constraints are especially useful in traffic networks. For example, they can be used to model arc capacities, which has been recognized as an important means to provide a more accurate description of traffic flows. Moreover, real-world transportation models often include such constraints, not only to prevent arcs from carrying arbitrarily large flows, but also as a way to calibrate the network model and bring its solution into agreement with anticipated results. In this more general model, multiple user equilibria might exist and an arbitrary user equilibrium does not need to be close to a system optimum. Nevertheless, we characterize a particular equilibrium that we call *Beckmann user equilibrium*, which turns out to be efficient. Defining the *coordination ratio* of a solution as the ratio between its total travel time and that of the system optimum, we show that the coordination ratio of a Beckmann user equilibrium is not higher than the price of anarchy in the model without side constraints. Beckmann user equilibria arise from an extension of the mathematical programming formulation used to compute user equilibria in the model without side constraints, due to Beckmann, McGuire, and Winsten (1956). Besides having relatively small total travel time, Beckmann user equilibria are attractive because they can be computed efficiently. This also shows that game-theoretic concepts successfully offer a way of computing approximate solutions to hard problems. Although there are multiple equilibria and computing the best equilibrium is difficult, a provably good user equilibrium can be computed in polynomial time.

We have just argued that, in the worst case, equilibria are not too inefficient. But what happens in the average case? Or at least in real-world examples? It turns out that equilibria are even closer to system optima, which is good news. But not quite. A number of studies with realistic instances coming from road traffic and telecommunications networks concluded that equilibria are of the order of 10% more costly than system optima (see Section 2.5). The Urban Mobility Study, a survey conducted by the Texas Transportation Institute (2002), recently estimated that the *congestion bill* in the U.S. alone was $67.5 billion in 2000, consisting of 3.6 billion hours of delay that people lose to highway congestion plus 5.7 billion gallons of gas. Given those figures, even a small improvement in the efficiency of the road traffic system implies that a lot of money (and time) could be saved. Therefore, in certain situations, a system manager might want to do better than in equilibrium.

Transportation officials hope that *Intelligent Transportation Systems* can help to alleviate the problem of congestion caused by the ever increasing amount of road traffic.[1] Among the proposed Intelligent Transportation Systems that have emerged in the last decade, so-called route guidance devices are an especially promising technology. In such systems, visual and acoustic indicators connected to the device guide drivers to their destinations, which they have specified at the beginning of the trip. The importance of route guidance comes from the fact that it can not only provide mapping information to drivers, but it could also be used as a mechanism to provide congestion-related information. As we said before, guiding individual users according to a system optimum is not an option because it is unfair to them. We explore a novel approach to route guidance by explicitly aiming at a traffic assignment that minimizes the total travel time in the network but with the incorporation of *user constraints*. Namely, when deciding on recommendations for users, we do not even consider routes that are too long because users are not likely to follow them even if recommended. Our approach offers significant advantages: On the one hand, it guarantees greater fairness for the individual user compared to the system optimum. On the other hand, the total travel time in the proposed solution is still close to that of the system optimum and thus in general better than in a user equilibrium. In addition to formulating the problem, we develop an algorithm for its solution that is efficient in practice. Not only does this show that the route guidance approach is feasible, the implementation also allows us to test real-world instances. Besides performing a detailed computational analysis, we also conduct a theoretical study of the parameters of the model to guarantee that the solutions we compute are indeed efficient and fair. To theoretically evaluate the efficiency of such a system, we compare the proposed solution to the situation without guidance, that is, the user equilibrium.

In other situations, the network authority may want to use a different objective. Instead of minimizing the total travel time, the authority might wish to minimize the

[1] However, some are pessimistic. For example, the same traffic study forecasts that “The bad news is that even if transportation officials do all the right things, the likely effect is that congestion will continue to grow” (Texas Transportation Institute 2002).

maximum delay, i.e., minimize the longest flow-carrying path in the network. A straightforward application of this objective is given by evacuation problems in which people have to leave an area struck by disaster as soon as possible. Some previous research has examined algorithms to compute an optimal evacuation pattern in a dynamic network. In a chaotic situation, it can be difficult to enforce a pre-computed optimal solution. Therefore, it is a better idea to design systems in which users act naturally as in an approximately optimal solution. Other important applications of this objective can be found in the study of computer networks, where it is desirable for the maximum latency of transmissions to be small, and in supply chain management, where a product can be assembled only when all the components have arrived at the production facility. To the best of the author's knowledge, this is the first work that considers minimizing this objective in static networks with congestion.

We show that the optimal solution with respect to the maximum latency objective is NP-hard to compute, even in the case of a single source, a single sink, and linear latencies. Again, we show that the price of anarchy of this system is bounded, although now with respect to this new objective. This implies the existence of a simple approximation algorithm, because computing the user equilibrium flow can be done in polynomial time. (Computing the system optimum is also an approximation algorithm for this problem.) Conversely, the optimal solution with respect to the maximum latency objective has small total travel time. Surprisingly, it is also fair when there is a single source, a single sink, and latency functions are linear.

1.1 Outline and Main Contributions

This section presents an outline of this work and an overview of the main contributions. The dissertation is divided into seven chapters, including the current one and the conclusions at the end. It is advised to read them in order because all of them draw on concepts from earlier chapters; the only exceptions may be Chapters 3 and 4, which are independent of each other. We explicitly point out such relations in the chapter-by-chapter discussion that follows.

Chapter 2: Preliminaries

Chapter 2 offers a review of the central concepts needed for the subsequent chapters. We present our network model, fundamental concepts such as the system optimum and the user equilibrium, and well-known results related to those concepts. We also discuss some simple examples to illustrate the model and the results. Finally, we review the central concept of the price of anarchy, which is used to measure the efficiency of systems controlled by independent and selfish individuals. This chapter is a prerequisite for all the chapters that follow.

Chapter 3: The Price of Anarchy for Networks with Side Constraints

Chapter 3 considers networks with convex side constraints on the vector of arc flows and introduces a definition of equilibrium that relies on user behavior. Side constraints are important for many applications. Moreover, they can be used to incorporate relations that are known to hold in the real-world. After comparing our concept of equilibrium with other definitions, we study the properties of these equilibria. In contrast to the basic model that we present in Chapter 2, the model with side constraints might have multiple equilibria. We therefore analyze the coordination ratio of the best and the worst equilibrium. Finally, we consider lower semicontinuous functions, the most general class of latency functions that guarantees the existence of user equilibria.

Main Contribution: Recent results show that in a transportation network with congestion (but without side constraints), the coordination ratio of an instance can be bounded by a constant (Roughgarden and Tardos 2002; Roughgarden 2003b). This constant does not depend on the network topology or demand structure but only on the allowable class of latency functions. We show that side constraints do not matter if one looks at equilibria from the right perspective. For instance, although there might be multiple equilibria and the coordination ratio of the worst equilibrium might not be bounded, the worst-case coordination ratio of the best equilibrium coincides with the price of anarchy in the model without side constraints. Moreover, the assumptions we make on latency functions are relatively mild. In contrast to previous work, they are not required to be *convex*, *differentiable*, or even *continuous*. These improvements result from a simplified proof based on a variational inequality approach.

Bibliographic Information: This chapter is based on a research article by Correa, Schulz, and Stier-Moses (2004b). A preliminary version appeared in Schulz and Stier-Moses (2003).

Chapter 4: An Efficient Route Guidance System

In Chapter 4, we describe our route guidance model and give details of the algorithm to compute associated solutions. Furthermore, we show how to calibrate the various parameters of the model and provide comprehensive computational results based on real-world networks that show that the model provides good solutions. This chapter is based on the model that we introduce in Chapter 2, and it can be read independently of Chapter 3 because we do not consider the side constraints introduced therein.

Main Contribution: The goal of the route guidance model introduced in this chapter is minimizing the travel time in the network subject to user constraints. These user constraints prevent users from being assigned to routes that they might regard as too long. For instance, we constrain users to not travel more than φ times their respective shortest paths, where φ represents a guess of the users' tolerance to being assigned to sub-optimal routes. We assume that φ is fixed and given in advance. The route lengths in the constraints are computed with respect to a measure of arc lengths given a priori that we call *normal lengths*. We propose different ways for selecting normal lengths. For instance, we consider *free flow* travel times, and travel times when the prevailing condition is a user equilibrium. We show how to compute the optimal solution to the route guidance problem and comment on the software that we implemented for that purpose. If normal lengths are user equilibrium travel times, then the proposed system yields substantial improvements in practice. For several medium to large real-world instances, we compute traffic assignments of notably smaller total travel time than in a user equilibrium; at the same time, they possess fairness attributes unrivaled by the ordinary system optimum. In contrast, selecting free flow travel times as normal lengths does not turn out to provide efficient solutions.

Bibliographic Information: This chapter is based on a research article by Jahn, Möhring, Schulz, and Stier-Moses (2004).

Chapter 5: Efficiency and Fairness of the Route Guidance System

Chapter 5 presents analytical worst-case bounds for the efficiency and fairness of solutions computed by the route guidance system that we present in Chapter 4. To measure the improvement in efficiency if the route guidance system is adopted, we compare the traffic patterns it returns to a user equilibrium. That allows us to determine the theoretical efficiency of the system when normal lengths are given by free flow travel times or user equilibrium travel times. Finally, we present upper bounds for the unfairness of optimal solutions to the route guidance problem. This chapter also draws strongly on the concepts related to the price of anarchy developed in Chapter 3.

Main Contribution: For free flow normal lengths, we show that if the tolerance φ is too small, user equilibria outperform the solutions returned by the route guidance system, which defeats the purpose of such a system. However, when normal lengths are user equilibrium travel times, the opposite is true. Indeed, we can prove that the solutions to the route guidance problem are always more efficient than user equilibria.

The route guidance method that we propose is not *incentive compatible*, which means that some users might have reasons to deviate from the system recommendation and select a different route. This happens because the system does not provide the shortest route to each user; otherwise, the solution would be a user equilibrium. Therefore, some users might not comply with the recommendations given by the route guidance system. We quantify to what extent this poses a problem by presenting bounds for the unfairness of the solutions proposed by it.

Bibliographic Information: This chapter is based on an extended abstract by Schulz and Stier-Moses (2003).

Chapter 6: The Maximum Latency Problem

In Chapter 6, we examine a different objective: minimizing the maximum latency experienced by users. We study the complexity of computing the optimal solution with respect to that objective and the characteristics of an optimal solution (which we call a min-max flow). In addition, we design constant-factor approximation algorithms for this problem. Conversely, we study approximation guarantees of the min-max flow with respect to the other objectives. Although independent from previous chapters, this chapter relates to various concepts introduced before.

Main Contribution: We study the flow that minimizes the maximum latency that users experience. In particular, we examine its relation to system optima (solutions of minimal total travel time) and user equilibria, with respect to the different performance measures that we consider (total travel time, fairness, and maximum latency). To that effect, we present tight worst-case bounds for the three mentioned solutions with respect to those three objectives. For single-source single-sink instances with linear latencies, we establish that all users experience the same delay in a solution that minimizes the maximum latency. In addition, we prove that the problem is NP-hard even with linear latency functions. For the single-source single-sink case, the price of anarchy with respect to the maximum latency coincides with the price of anarchy with respect to the total latency. This result is important because it implies that computing a user equilibrium (which can be done in polynomial time) represents an approximation algorithm for the problem. Contrary to the conclusions of Chapter 3, the lack of central coordination has more severe consequences when multiple commodities are present. Indeed, the price of anarchy is unbounded even with linear latencies. With regards to fairness, we present a family of examples that shows that in optimal solutions to instances with nonlinear latency functions some users have to travel arbitrarily more than others.

Finally, another objective that we study is the maximum latency of an arc. In contrast to the objective of minimizing the maximum latency experienced by users, an optimal solution with respect to this objective can be arbitrarily bad when measured with respect to the three objectives mentioned in the previous paragraph. Conversely, system optima, user equilibria, and min-max flows can be arbitrarily bad with respect to the maximum arc-latency objective.

Bibliographic Information: This chapter is based on a research article by Correa, Schulz, and Stier-Moses (2004a).

1.2 Comparison with Other Work

Table 1.1 compares our results with others that are available in the literature. Although in many cases there are follow-up articles to those we indicate, for the sake of simplicity we include only the initial paper. We provide details of the models and mention additional articles in the corresponding chapters. We restrain ourselves from presenting other research related to the price of anarchy here because it is not based on the model we use. In Chapter 7, however, we take a step back to discuss the broader picture, and that is a good occasion to mention those articles.

The table is divided in four parts as follows. The first group contains work related to the initial result concerning the price of anarchy (Koutsoupias and Papadimitriou 1999). Those articles consider a simple network consisting of parallel arcs that connect a single source to a single sink, and the social goal is to minimize the maximum latency. In this model, flow is controlled by finitely many users, and each user may only select a single arc to route her flow. For more details concerning this model, we refer the reader to Section 2.5. The second part groups results that consider the model of Roughgarden and Tardos (2002), which assumes an arbitrary multicommodity flow network with infinitely many users, each controlling an infinitesimal amount of flow. The social goal is the total latency of a flow. Chapter 3 includes a detailed comparison of the results within this group. The third group lists results that bound the unfairness of flows. As Roughgarden's result (2003a) and ours were obtained independently and at the same time, we list both. We provide details of this group of results in Chapters 5 and 6. Finally, the fourth group contains results about the maximum latency of a flow in arbitrary networks. Although these results use the same objective as those in the first group, they consider divisible flow that is controlled by infinitely many users; see Chapter 6 for details.

Table 1.1: Comparison with other work

Reference [a]	Model [b]	Latencies [c]	Objective [d]	Constraints [e]	Users [f]
KP99	par	linear	max/bn	none	uns
BGGM03	par	linear	tot	none	uns
FKS04	st [g]	linear	max/tot	none	uns
RT02	netw	linear	tot	none	inf [h]
Rou03b	netw	cd	tot	none	inf
SS03	netw	cd	tot	cap	inf
Chapter 3	netw [i]	lsc	tot	sca [j]	inf
KK03	netw	linear	tot	scp	inf
RT04	ncg	cd	tot	none	inf
CS03	ncg [k]	sym	tot	none	inf
Per04	netw	asym	tot	none [l]	inf
Rou02	st	cd	unf	none	inf
Rou03a	netw	diff	unf	none	inf
Chapter 5	netw	diff	unf	none	inf
Wei01	st/netw	cd	max	none	inf
Chapter 6	st/netw	multiple [m]	max/bn	none/sca	inf
Rou04	st	cont	max	none	inf

[a] This column references the following papers: [BGGM03] Berenbrink et al. 2003, [CS03] Chau and Sim 2003, [FKS04] Fotakis, Kontogiannis, and Spirakis 2004, [KK03] Karakostas and Kolliopoulos 2003, [KP99] Koutsoupias and Papadimitriou 1999, [Per04] Perakis 2004, [Rou02] Roughgarden 2002, [Rou03a] Roughgarden 2003a, [Rou03b] Roughgarden 2003b, [Rou04] Roughgarden 2004b, [RT02] Roughgarden and Tardos 2002, [RT04] Roughgarden and Tardos 2004, [SS03] Schulz and Stier-Moses 2003, [Wei01] Weitz 2001.

[b] Models: [par] single-origin, single-destination, and parallel links, [st] s-t-network with arbitrary topology, [netw] multicommodity and arbitrary network, [ncg] nonatomic congestion game.

[c] Latency functions are always nonnegative, nondecreasing, and separable unless otherwise noted. In addition, they were considered to be: [linear] linear functions (including a constant term) [cd] s-convex and differentiable functions, [diff] differentiable, [cont] continuous, [lsc] lower semicontinuous, [sym] continuously differentiable, non-separable, and symmetric, [asym] differentiable, non-separable, and positive semidefinite Jacobian.

[d] Objectives that were considered: [bn] bottleneck (maximum arc-latency), [max] maximum latency, [tot] total latency, [unf] unfairness.

[e] Constraints allowed in the problem: [none] no constraints, [cap] capacity constraints, [sca] side constraints on arc flows, [scp] side constraints on path flows.

[f] Assumptions concerning the number of users in the system and whether flow can be split: [inf] infinitely many users each controlling an infinitesimal of the flow, [uns] finitely many users who cannot split their flow.

[g] Positive results assume a rather restricted class of networks.

[h] Although it is not their main model, they also consider [uns], and finitely many users who can split their flow.

[i] Can be extended to [ncg]; see Section 3.9.

[j] Can also be extended to [scp]; see Section 3.9.

[k] In addition, this article also considers a model with elastic demands.

[l] Her results can be extended to include arbitrary side constraints; see Section 3.9.

[m] We make different assumptions for each result. For details, we refer the reader to Chapter 6.

Chapters 4 and 5 propose a new model for route guidance and, therefore, cannot be easily compared to earlier work. For that reason, we did not include those results in Table 1.1. Let us note, however, that Holmberg and Yuan (2003) independently developed a model similar to ours to compute routing patterns for telecommunication networks. They propose to compute a solution with smallest total latency among those that satisfy time-delay or reliability requirements. Although both models and the respective solution approaches have some similarities, their goals are totally different from ours.

1.3 About this Dissertation

This dissertation is not meant as an introduction to network flows, optimization, or game theory. Nevertheless, we do not make many assumptions concerning the reader's prior knowledge, except that the reader possesses a basic mathematical maturity. For in-depth introductions to those topics, we refer the reader to the following sources. The books by Sheffi (1985), Nagurney (1993), and Patriksson (1994), and the articles by Magnanti (1984), and Florian and Hearn (1995) provide an introduction to traffic flow theory and traffic equilibria. The book by Bertsekas and Gallager (1991) introduces similar concepts from the perspective of telecommunication networks. The book by Bertsekas (1999) is a good introduction to nonlinear programming, and the books by Ahuja, Magnanti, and Orlin (1993), and Bertsekas (1998) on network optimization complement the ones cited earlier. Another useful reference is Garey and Johnson's (1979) introduction to computational complexity and the theory of NP-completeness. Finally, the books by Fudenberg and Tirole (1991), Osborne and Rubinstein (1994), and Owen (1995) are good introductions to game theory.

We make every effort to define all the concepts that we use. For convenience, we list the notation on the margin and in the index, and we use *italics* to denote names used for the first time.

Chapter 2

Preliminaries

In this chapter we lay the ground work for this thesis. Section 2.1 presents the specifics of the model together with the obligatory notation. The models we shall discuss in the following chapters are different generalizations of our basic model. We illustrate the network model with instances, which will be repeatedly used later on. Section 2.5 presents the concept of the *price of anarchy*, one of the key notions that we use extensively. Finally, and to show the power of our approach, we prove a result about the price of anarchy for the basic model. Although this result is already known, we are able to give a simpler proof relying on a variational inequality that characterizes a user equilibrium.

2.1 The Basic Traffic Model

This section introduces the concepts and notation that will be used throughout this dissertation. We start by defining the elements of an instance of a network with congestion. We represent the network by a directed multigraph $G = (N, A)$, where N is the set of nodes and A is the set of arcs in the network. In addition, the set $K \subseteq N \times N$ of origin-destination (OD) pairs models the demand. For each $k = (s_k, t_k) \in K$, a flow of rate $d_k > 0$ must be routed from the origin s_k to the destination t_k. In the context of traffic networks, it is typical to assume infinitely many users and that each individual controls only an infinitesimal fraction of the flow. For $k \in K$, let $\mathcal{P}_k$ be the set of

G, N

A, K

d_k

$\mathcal{P}_k$

$\mathcal{P}$

directed (simple) paths in G connecting the corresponding origin with its destination, and let $\mathcal{P} \overset{\text{def}}{=} \bigcup_{k \in K} \mathcal{P}_k$. For certain results, we shall assume that there is a single origin and a single destination. We sometimes refer to those networks as *s*-*t*-networks and to their corresponding flows as *s*-*t*-flows.

$\ell_a(x_a)$

s-convex

Congestion is modeled by *latency functions*[1] associated with the arcs. For each arc $a \in A$, there is a nonnegative and nondecreasing latency function ℓ_a with values in $\mathbb{R}_{\geqslant 0} \cup \{\infty\}$ that maps the flow on arc a to the time needed to traverse a. These latency functions are called *separable* because the arc traversal time depends only on the flow along the same arc. For some results in this thesis, additional assumptions are needed; we explicitly indicate them when they are required. One of the assumptions that we sometimes make is that $x_a\, \ell_a(x_a)$ is a convex function; in that case we say that ℓ_a is *s-convex* (Bergendorff, Hearn, and Ramana 1997).

$\mathcal{L}$

We only consider latencies that belong to a set $\mathcal{L}$ specified in advance. The reason is that, in practice, latency functions take a specific form, and the bounds we would obtain without any restriction are pessimistic (Roughgarden and Tardos 2002). The only assumptions we make on $\mathcal{L}$ are that it includes a constant function and that it is closed under multiplication (implying that it includes all constants). We do not assume that it is closed under addition but that is irrelevant because one can always add latency functions by subdividing an arc. The network G together with the arc latencies ℓ_a for $a \in A$ and the traffic to be routed—represented by K and $(d_k)_{k \in K}$—is called an *instance* of the traffic routing problem.

$\mathcal{L}_{\text{lin}}$

To make the analysis more tractable, for some results we will assume that the latencies are linear functions,[2] i.e., they belong to $\mathcal{L}_{\text{lin}} \overset{\text{def}}{=} \{\ell : \ell(x) = qx + r : \text{ for some } q, r \geqslant 0\}$. Although this may appear restrictive at first sight, congestion and counterintuitive phenomena may already be present with linear latencies. A paradox named after Braess that is presented in Section 2.4.2 shows how intuition might sometimes be misleading.

Latency functions used in practice are usually smooth and convex (Cohen 1991;

[1] Latency functions are also known as *link performance functions*.

[2] To be more precise, we should say 'affine' because the functions we refer to contain an additive constant term. However, as they have been usually referred to as 'linear,' we prefer to continue with the same denomination.

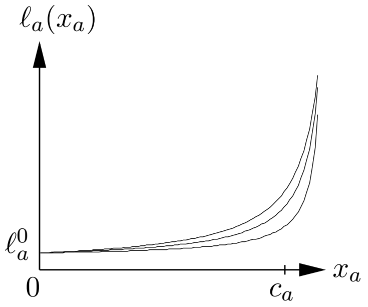


Figure 2-1: Typical link delay functions

Sheffi 1985). Thus, they satisfy the assumptions we just mentioned as well as those that we will mention later. Figure 2-1 illustrates the typical shape of these functions: after they reach the *practical capacity* c_a (Patriksson 1994), they grow very fast. In practice, a c_a commonly used function is that put forward by the U.S. Bureau of Public Roads (1964):

$$\ell_a(x_a) \overset{\text{def}}{=} \ell_a^0 \cdot \left(1 + \zeta \left(\frac{x_a}{c_a}\right)^\eta\right) , \tag{2.1}$$

where $\ell_a^0 > 0$ is the travel time in the uncongested network (also called *free flow* travel time) and $\zeta \geqslant 0$ and $\eta \geqslant 0$ are tuning parameters. However, more general latency functions might be needed on certain occasions. For example, some proposed congestion pricing schemes make use of step-function tolls (Bernstein and Smith 1994), modeled with discontinuous latency functions.

A *path flow* (or flow on paths) is a nonnegative vector $x = (x_P)_{P\in\mathcal{P}}$ that meets the x demand, i.e., $\sum_{P\in\mathcal{P}_k} x_P = d_k$ for all $k \in K$. Given a path flow, the corresponding *arc flow* (or flow on arcs) is easily computed as $x_a = \sum_{P\ni a} x_P$, for each $a \in A$. Often, we shall only write 'flow' and the meaning should be understood from the context. For a flow x, the travel time along a path P is $\ell_P(x) \overset{\text{def}}{=} \sum_{a\in P} \ell_a(x_a)$. Hence, the flow's total $\ell_P(x)$ travel time (or latency) is

$$C(x) \overset{\text{def}}{=} \sum_{P\in\mathcal{P}} \ell_P(x) x_P = \sum_{a\in A} \ell_a(x_a) x_a \,. \tag{2.2}$$ $C(x)$

In addition, we also care about the average latency because that quantity is comparable to the maximum latency. We let the average latency of a flow x between OD pair $k \in K$

$\overline{C}_k(x)$ be defined as $\overline{C}_k(x) \overset{\text{def}}{=} \sum_{P \in \mathcal{P}_k} \ell_P(x) x_P / d_k$, and the average travel time of x be

$\overline{C}(x)$

$$\overline{C}(x) \overset{\text{def}}{=} \frac{\sum_{P \in \mathcal{P}} \ell_P(x) x_P}{\sum_{k \in K} d_k} .$$

$L_k(x)$ $L(x)$ Finally, we denote the maximum latency of a flow x for an OD pair $k \in K$ by $L_k(x) \overset{\text{def}}{=} \max\{\ell_P(x) : P \in \mathcal{P}_k,\ x_P > 0\}$, and the maximum latency among all users by $L(x) \overset{\text{def}}{=} \max_{k \in K} L_k(x)$.

2.2 The System Optimum

Wardrop (1952) enunciated two principles that have been extensively used to describe traffic flows. For convenience, let us start by describing the second one. The principle says that a network manager should aim at a solution that minimizes the total travel time. To achieve that, the manager could solve a mathematical program whose optimal solution satisfies the mentioned principle.

SO **Definition 2.1.** A system optimum f^* is an optimal solution to the following nonlinear minimum cost multicommodity flow problem with a separable objective function:

$$\text{Problem SO:} \quad \begin{array}{lllr} \min & C(x) & & (2.3\text{a}) \\ \text{s.t.} & \sum_{P \in \mathcal{P}_k} x_P = d_k & k \in K, & (2.3\text{b}) \\ & x_P \geqslant 0 & P \in \mathcal{P}. & (2.3\text{c}) \end{array}$$

f^* We will denote system optimal flows by f^*. Note that although this formulation has exponentially many variables, the same problem can be written with arc variables using the standard flow conservation constraints. Knowing an optimal flow on arcs permits us to recover a flow on paths easily; any flow decomposition works. Therefore, if $C(x)$ is convex, this problem can be solved in polynomial time by invoking the ellipsoid method[3]

[3] Note that the ellipsoid method, as well as interior point methods, can only solve the problem up to an arbitrary small additive error term. As the example given in Section 3.4.1 shows, an optimal solution may need irrational numbers, which cannot be represented exactly.

(see, e.g., Vavasis 1991; Grötschel, Lovász, and Schrijver 1993).

Beckmann, McGuire, and Winsten (1956) proved that first-order optimality conditions can be used to easily characterize the optimal solution to this problem (see also Dafermos 1972; Bertsekas 1999 is a good introduction to optimality conditions in convex optimization problems). Indeed, under the same convexity and differentiability assumptions, first-order optimality conditions imply that a flow f^* is optimal if and only if

$$\ell^*_P(f^*) \leqslant \ell^*_Q(f^*) \quad \text{for all } k \in K \text{ and all paths } P, Q \in \mathcal{P}_k \text{ such that } f^*_P > 0\,. \tag{2.4}$$

Here, the modified travel time $\ell^*_a(x_a) \overset{\text{def}}{=} (\ell_a(x_a)x_a)' = \ell_a(x_a) + \ell'_a(x_a)x_a$ for an arc $a \in A$ includes an extra term that accounts for *external costs*[4] users cause to others. Condition (2.4) is very similar to one that describes flows at equilibrium, which we now proceed to define. This similarity will turn out to be very useful. $\ell^*_a(x_a)$

2.3 The User Equilibrium

A key assumption that motivates the current work is user selfishness. This means that users only consider (or care about) their individual utilities. In our model, their goal is minimizing the delay they experience when traveling from their origins to their destinations (Kohl 1841). As long as there are better routes, users will update their choices and select shorter paths with respect to their previously experienced condition. This process converges to an equilibrium, in which users are distributed along the network in a way such that each and every one travels along shortest paths under the prevailing congestion condition (Friesz et al. 1994; Larsson and Patriksson 1999). This concept is exactly what Wardrop (1952) indicated in his first principle.

Definition 2.2. A feasible flow f is a *Wardrop equilibrium* if

$$\ell_P(f) \leqslant \ell_Q(f) \quad \text{for all } k \in K \text{ and all paths } P, Q \in \mathcal{P}_k \text{ such that } f_P > 0\,. \tag{2.5}$$

[4] The *externalities* are the additional delay that users impose to others by their presence in the system: users generate congestion that have a negative impact on others' utilities.

At around the same time, Nash (1951) published an influential paper that defined an equilibrium as a solution to a game in which no player has any incentive to deviate. This motivates the following definition.

Definition 2.3. A feasible flow f is a *Nash equilibrium* if no small bundle of flow has any incentive to switch to another path. Formally, for all $k \in K$, all paths $Q, R \in \mathcal{P}_k$ such that $f_Q > 0$, and all $0 \leqslant \varepsilon \leqslant f_Q$, if we let a new flow f^ε be defined by

$$f^\varepsilon_P \overset{\text{def}}{=} \begin{cases} f_Q - \varepsilon & \text{if } P = Q, \\ f_R + \varepsilon & \text{if } P = R, \\ f_P & \text{otherwise,} \end{cases} \qquad \text{for } P \in \mathcal{P}\,,$$

we must have that $\ell_Q(f) \leqslant \ell_R(f^\varepsilon)$.

A flow satisfying the preceding definition is a Nash Equilibrium of a non-cooperative game among the network users. In this game, users wish to select a route with minimal travel time. Recall that we assume that an infinite number of players participate in the network game, and each controls an infinitesimal amount of flow. Although we only allow players to select *pure* strategies (paths) as opposed to *mixed* strategies (distributions over paths), this makes no essential difference because each player controls a negligible amount of flow. Haurie and Marcotte (1985) discussed the difference of using pure and mixed strategies formally and showed that a pure Nash equilibrium of a game in which the number of players tends to infinity approaches the Nash equilibrium as in Definition 2.3. We must mention that there exist related results in the domain of telecommunication networks in which equilibria are also used to characterize traffic. However, a key difference is the consideration of a finite number of users, each controlling a non-negligible amount of flow (see, e.g., Orda, Rom, and Shimkin 1993).

Definitions 2.2 and 2.3 originated in different communities. Actually, the connection between Wardrop and Nash equilibria was first made by Charnes and Cooper (1961). As it turns out, they are equivalent in network games with continuous and nondecreasing latency functions (see, e.g., de Palma and Nesterov 1998).

Proposition 2.4. *A feasible flow f of an instance that has continuous and nondecreasing latency functions is a Nash equilibrium if and only if f is a Wardrop equilibrium.*

The two assumptions of Proposition 2.4 are necessary: Wardrop and Nash equilibria might differ (or even not exist) if any of the two assumptions is relaxed. Because all users travel along shortest paths, the cost of an equilibrium can be simply expressed as $C(f) = \sum_{k\in K} L_k(f)d_k$. As latencies used in practice are nondecreasing and often continuous, previous work has not generally differentiated between the two equilibria and referred to flows at equilibrium as *user equilibria* (Dafermos and Sparrow 1969). UE
For this reason, we choose to refer to flows satisfying Definitions 2.2 and 2.3 as user equilibria throughout the dissertation, and we denote them by f. We refer the reader to f
Bernstein and Smith (1994), and de Palma and Nesterov (1998), who described in detail the types of equilibria used in traffic models and their differences (see also Section 3.8).

Let us remark that the simplicity of these concepts (and their tractability) prompted transportation practitioners to adopt user equilibria as a way to model and predict user behavior. For instance, user equilibria have been widely used in traffic planning projects around the world as the main analytical tool.

Not surprisingly, the total (or equivalently, average) travel time is generally not minimized by the user equilibrium, since users do not pay for their external costs (Dupuit 1849; Pigou 1920; Knight 1924). Transportation economists have long proposed collecting tolls from the users of the network to obtain an efficient equilibrium (see, e.g., Beckmann, McGuire, and Winsten 1956; Arnott and Small 1994; Bergendorff, Hearn, and Ramana 1997; Transport for London 2004 reports on a simple toll pricing scheme recently implemented in London, UK). If in every arc travelers pay for the delay they impose to others by their presence, it can be shown that an equilibrium is a system optimum of the original instance. In other words, f^* is optimal if and only if the marginal travel time of any used path is not greater than that of any other path. The last sentence can be interpreted as the following proposition, which is implied by Condition (2.4).

Proposition 2.5 (Beckmann, McGuire, and Winsten 1956). *Let f^* be a feasible* *flow for an instance with nondecreasing, s-convex, and differentiable latency functions. Then, f^* is a system optimum with respect to latencies ℓ if and only if f^* is a user equilibrium with respect to latencies ℓ^*.*

Note that s-convexity is only needed to show one of the implications. Namely, we would not be able to show the backward implication because the solution would only be a local optimum. The previous proposition provides the fundamental idea of toll pricing: if users are charged a constant toll equal to $\ell'_a(f^*_a)f^*_a$ in every arc, the resulting user equilibrium is a system optimum.

Notice the similarity of Conditions (2.4) and (2.5). This relation was successfully exploited by Beckmann, McGuire, and Winsten (1956), who proposed to compute a user equilibrium by modifying Problem SO. Indeed, (2.5) can be interpreted as the optimality conditions of a convex min-cost multicommodity flow problem similar to Problem SO, but with the objective in Equation (2.3a) replaced by

$$\sum_{a \in A} \int_0^{x_a} \ell_a(y)dy \,. \tag{2.6}$$

This transformation works because the derivative of $\int_0^{x_a} \ell_a(y)dy$ with respect to x_a is precisely $\ell_a(x_a)$. This is the essence of what is needed to prove the following theorem.

Theorem 2.6 (Beckmann, McGuire, and Winsten 1956). *Consider an instance with continuous and nondecreasing latency functions. A user equilibrium always exists, it is essentially unique and it can be computed efficiently using standard procedures.*

Here, *essentially unique* means that under different equilibria each user experiences the same travel time. More formally, if f and f' are user equilibria, $L_k(f) = L_k(f')$ for all $k \in K$. Of course, this implies that all equilibria share the same total cost. When latencies are strictly increasing, the objective is strictly convex and, in that case, the equilibrium as a flow on arcs is unique. Note that the convexity of $C(x)$ is not required in the last theorem; continuity and monotonicity are enough to guarantee that the objective function displayed in (2.6) is convex. Thus, in the space of flows, the set of all equilibria is nonempty and convex. Furthermore, a user equilibrium can be computed in polynomial time by solving the flow problem previously described (Problem SO with the convex objective function shown in (2.6) and formulated with arc variables). An optimal solution to that problem is an equilibrium encoded as a flow on arcs. Finally, a path representation can be obtained via any flow decomposition of that optimal solution; it is guaranteed that all users of the same OD pair will experience the same travel time. Note that the approach is similar to computing a system optimum; see the comment after Definition 2.1.

For a more detailed discussion concerning the existence, uniqueness, algorithmic techniques, and related aspects of equilibria, we refer the reader to Dafermos (1980), Magnanti (1984), Friesz (1985), Sheffi (1985), Florian (1986), Nagurney (1993), Patriksson (1994), and Florian and Hearn (1995). Note that many generalizations of the basic model have been considered. To mention a few examples, there are extensions to multiple modes of transit, link interactions, and demand relationships (Dafermos 1972; Florian 1977; Aashtiani and Magnanti 1981).

Because of its simplicity and power, we are particularly interested in an equivalent characterization of user equilibria in terms of a variational inequality problem due to Smith (1979); see also Dafermos (1980). Let us define another cost function that will play an important role in Chapter 3. We fix a given feasible flow f. For an arbitrary feasible flow x, its cost with respect to the flow f is

$$C^f(x) \overset{\text{def}}{=} \sum_{a \in A} \ell_a^f x_a \,, \tag{2.7}$$ $C^f(x)$

which is equivalent to computing the (standard) cost with respect to constant latencies $\ell_a^f \overset{\text{def}}{=} \ell_a(f_a)$. Notice that $C^f(f) = C(f)$. The following proposition is a direct consequence of the fact that at equilibrium, users travel on shortest paths with respect to arc costs ℓ_a^f.

Proposition 2.7 (Smith 1979). *A feasible flow f for an instance with continuous and nondecreasing latency functions is a user equilibrium if and only if*

$$C^f(f) \leqslant C^f(x) \quad \textit{for all feasible flows } x \,. \tag{2.8}$$

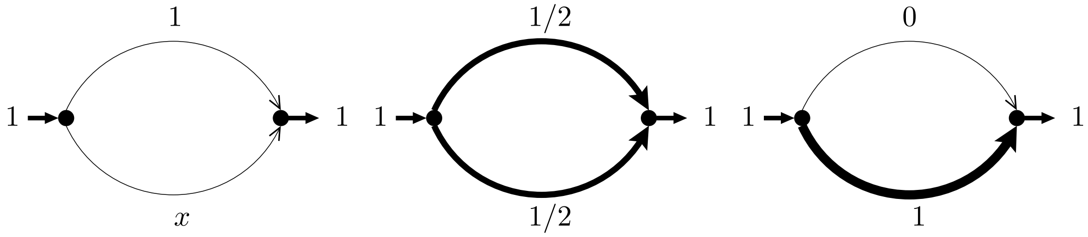


Figure 2-2: Pigou's example. *From left to right*: The instance (arcs are labeled with their latency functions), the system optimum (arcs are labeled with their flows), and the user equilibrium (idem).

2.4 Examples

In this section, we discuss two examples that will help the reader develop more intuition regarding the model we have just described. Furthermore, they will be the basis of various constructions to be developed in the following chapters.

2.4.1 Pigou's Instance

Perhaps the simplest possible example of an instance of the traffic model we just introduced was given by Pigou (1920) in his studies of economic markets; it is also described by Roughgarden and Tardos (2002). The example consists of a unit demand rate composed by an infinite number of users that have to travel between two terminals. Users can opt between two paths of different characteristics: the arc a on top, featuring a constant travel time, and the arc b on the bottom that has travel time equal to its flow. On the left side of Figure 2-2, we show the network that we have just described; arcs are labeled with their latency functions.

The system optimum can be calculated by solving the instance of Problem SO corresponding to this example:

$$\begin{aligned} \min \quad & x_a + x_b^2 \\ \text{s.t.} \quad & x_a + x_b = 1 \\ & x_a, x_b \geqslant 0\,. \end{aligned}$$

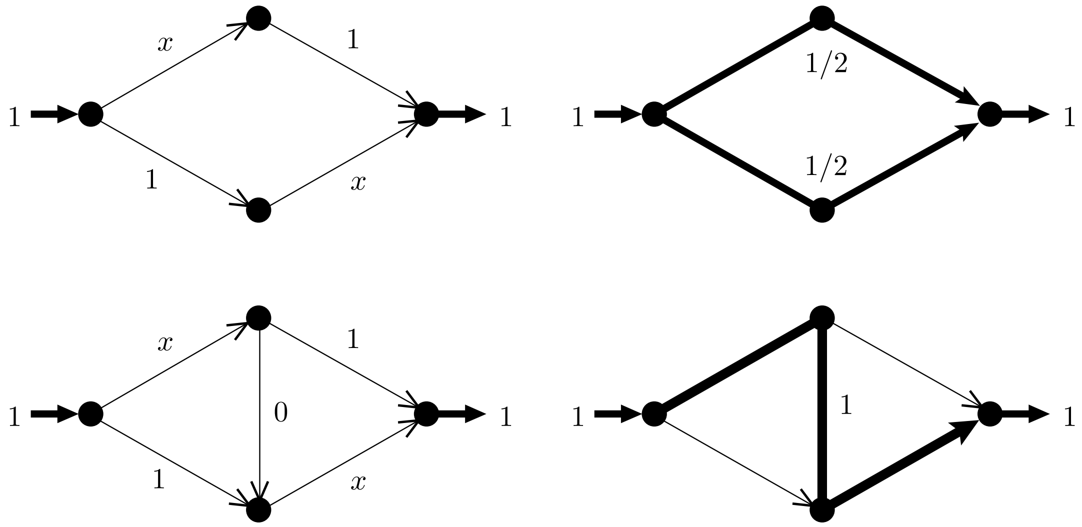


Figure 2-3: Braess' Paradox example. On the left we show the instances (arcs are labeled with their latency functions); on the right we show their user equilibria (arcs are labeled with their flows).

The optimal solution is $f^* = (1/2, 1/2)$, where the pair of values represents the flow on the top and the bottom arc, respectively (Figure 2-2, *middle*). The latencies of the arcs are $\ell_a(f_a^*) = 1$ and $\ell_b(f_b^*) = 1/2$, giving a total travel time $C(f^*)$ of 3/4 units of time. As the latency of the flow along arc a is 1 unit of time, while that of the flow along arc b is 1/2 unit, users on a will not be satisfied with the assignment and would rather switch to b.

The only solution in which no user has any incentive to switch is when all users take arc b (Figure 2-2, *right*). Indeed, that is a user equilibrium because all the users travel for one unit of time and the latency of arc a—the other option—is not smaller. The total travel time $C(f)$ equals one unit of time, which is obviously sub-optimal.

2.4.2 Braess' Paradox

Another relevant example is a seemingly paradoxical instance that was first presented by Braess (1968). The latency of *every* user in this network increases after a new arc is added to it. Therefore, adding a new arc increases the total travel time, contrary to what one might expect.

We start with the instance depicted in the top-left of Figure 2-3. The network, which contains two paths connecting a single origin to a single destination, has to route a unit demand. It is easy to see from the previous definitions that the system optimum and the user equilibrium coincide. Both route the flow by dividing it equally among the two paths, as shown on the top-right of the figure. It is easy to see that the travel time of every individual user and that of the whole system equal 3/2 units of time.

Consider "improving" the network by adding a new arc with latency function uniformly equal to 0. This new arc shortcuts the two paths as shown in the bottom-left of the figure. The system optimum does not change, but the user equilibrium does. Indeed, with respect to the previous user equilibrium, all users find that the path through the new edge is preferable (its latency is 1 unit of time vs. 3/2 units). Therefore, all users have an incentive to switch to the new path; the equilibrium is attained if all users are routed precisely along that path. At equilibrium, the latency of every user, as well as the total travel time, is 2.

The explanation of this apparent paradox is that there is no reason why users' latencies have to be monotone with network additions because nobody in such a system pays attention to the system-wide utility. The conclusion is that it is not always wise to construct new arcs if the planner's goal is to minimize the travel time in a network and route decisions are left to users. Note that before Braess, Downs (1962) predicted that under certain conditions, creating a new arc might make traffic congestion worse than before. However, the first concrete example is due to Braess. Among the vast amount of literature that followed Braess' article, Frank (1981) and Steinberg and Zangwill (1983) analyzed necessary and sufficient conditions for the existence of such paradoxical flows, and Hagstrom and Abrams (2001) did the same for a generalization of Braess' paradox. The results of Section 2.5.1 as well as the main result of Chapter 3 for general latency functions and networks with side constraints (Theorem 3.10) can be used to provide a worst-case bound on the degradation of the total (and therefore average) travel time that can possibly be caused by Braess' Paradox. Specifically, Roughgarden (2004a) and Lin, Roughgarden, and Tardos (2004) analyzed the worst-case severity of this effect. For s-t-networks, Roughgarden presented a tight upper bound for the improvement of the network performance after removing an arbitrary set of arcs. Subsequently, Lin, Roughgarden, and Tardos presented a stronger bound that depends on the number of arcs that are removed. Besides traffic networks, this paradox may also appear in other settings such as queuing and telecommunication networks (Cohen and Kelly 1990; Kameda et al. 2001).

2.5 The Price of Anarchy

The use of central coordination to achieve a system-wide objective is seldom feasible. It is usually unacceptable as users may not have an incentive to comply with the central directives. Although classic problems in operations research assume that there is a central authority that has the power to control the system, recently there has been a trend to acknowledge this difficulty, understand its consequences, and design systems that achieve coordination by other means. These models invariably include economic and game-theoretic aspects as a way to model user behavior. Some examples that have been or can be modeled from that perspective include the Internet, wireless networks, road traffic networks, transit networks, evacuation systems, distribution systems, auctions, and facility location problems, just to mention a few.

Koutsoupias and Papadimitriou (1999) proposed to measure the cost of lacking central coordination by comparing the cost of equilibria to that of an optimal solution. Although others had previously compared the performance of the best equilibrium to a system optimum (e.g., Papadimitriou and Yannakakis 1994, Shenker 1995, and Korilis and Lazar 1995), Koutsoupias and Papadimitriou argued that the comparison must be with respect to the worst equilibrium because without explicit control nobody can guarantee that the equilibrium that will turn out is any particular one. For a given instance of a problem and a corresponding solution, the ratio of its cost to the optimal cost is referred to as the *coordination ratio* of that solution. When a specific flow is not mentioned, we mean the coordination ratio of the worst-case equilibrium of the given instance. The worst-case coordination ratio among all the possible instances of a problem is referred to as the *price of anarchy* of the system (Papadimitriou 2001).

More concretely, Koutsoupias and Papadimitriou modeled a telecommunication network as two terminals connected by parallel links used to serve a finite number of communication requests between the terminals, each controlled by a different player. To measure the efficiency of a particular solution, the authors selected the maximal load of the links, a common objective in telecommunication network models. Among other things, they showed that the maximal load of an arc in an equilibrium is $\Omega(\log m/\log\log m)$ times that of an optimal solution, for m identical parallel links. They conjectured that the bound is tight, a fact subsequently proved by Mavronicolas and Spirakis (2001) for a particular case, and for the general one independently by Koutsoupias, Mavronicolas, and Spirakis (2003), and by Czumaj and Vöcking (2004). When the speed of the links is not uniform, the latter article proved that the worst-case ratio of the maximal load of an equilibrium to that of an optimal solution is $\Theta(\log m/\log\log\log m)$, a value slightly larger than that of the uniform case. Subsequently, Berenbrink et al. (2003), working with a model similar to that of Koutsoupias and Papadimitriou but with the total latency objective, showed that the price of anarchy is, again, bounded. Finally, Fotakis, Kontogiannis, and Spirakis (2004) extended some of these bounds to slightly more general networks, still with a single source, a single sink and finitely many players that cannot split their flow. It is surprising that these worst-case bounds do not depend on the number of connection requests or their rates. This fact is also true for the model we use in this dissertation, as we discuss below. For more details on these results, their extensions and related results, we refer the reader to the surveys by Feldmann et al. (2003b) and Czumaj (2004).

With a similar motivation, Roughgarden and Tardos (2002) studied the price of anarchy with respect to the total travel time. In contrast to the results discussed in the previous paragraph, they considered the model we described in Section 2.1 consisting of arbitrary networks with multiple OD pairs, infinitely many users, and infinitely divisible flows. The authors showed that the total travel time of an equilibrium is at most that of optimally routing twice as much traffic in the same network. Moreover, the total latency of selfish routing is at most 4/3 times that of the best coordinated routing, when the latency of every arc depends linearly on its congestion. Furthermore, Roughgarden (2003b) argued that the worst-case inefficiency due to selfish routing is independent of the network topology, where the inefficiency of an instance is measured by the coordination ratio. More specifically he proved the following result.

Theorem 2.8 (Roughgarden 2003b). *Let $\mathcal{L}$ be a family of nondecreasing, s-convex, and differentiable latency functions. Consider an instance of the traffic assignment problem with latency functions drawn from $\mathcal{L}$. Then, the ratio of the total travel time of a user equilibrium f to that of a system optimum f^* is bounded from above by the constant $\alpha(\mathcal{L})$. Moreover, this upper bound is tight.*

Note that the constant $\alpha(\mathcal{L})$ may be infinity depending on the choice of $\mathcal{L}$. Section 3.6 gives more details about Roughgarden's definition of $\alpha(\mathcal{L})$ and presents an alternative definition that is simpler and can be generalized to our more general setting. Surprisingly, instances for which the bound is tight are very simple: The constant $\alpha(\mathcal{L})$ turns out to be the coordination ratio of an instance similar to Pigou's (see Sections 2.4.1 and 3.6). Therefore, the price of anarchy for networks satisfying the theorem's assumptions is $\alpha(\mathcal{L})$. As in the model of telecommunication networks, the price of anarchy does not depend on the number of OD pairs, nor on the complexity of the network. Table 2.1 shows $\alpha(\mathcal{L})$ for a few classes of latency functions. For example, the first line of the table indicates that when all latency functions are monomials of the same degree but with arc-dependent nonnegative coefficients, user equilibria and system optima coincide (Dafermos and Sparrow 1969); if all latency functions are polynomials of degree four with arc-dependent nonnegative coefficients, the cost of the user equilibrium is not more than 2.151 times that of the system optimum.

While Theorem 2.8 only works for latency functions that are *nondecreasing*, *s-convex*, and *differentiable*, in Chapter 3 we generalize it to broader classes of functions and networks with side constraints on the vector of arc flows. We prove that the bounds of Table 2.1 are still valid so long as the correct equilibrium is selected. In contrast, the price of anarchy is unbounded because there exist inefficient equilibria.

Chapter 6 considers two generalizations of the objective selected by Koutsoupias and Papadimitriou: the maximum latency that users experience, and the maximum

Table 2.1: Guarantees for the efficiency of equilibria. All coefficients are assumed nonnegative.

Latencies are ... ($\mathcal{L}$)	Example	Price of Anarchy $\alpha(\mathcal{L})$
monomials of degree n	ax^n	1
linear functions	$a_1x + a_0$	4/3
quadratic functions	$a_2x^2 + a_1x + a_0$	1.626
cubic functions	$a_3x^3 + a_2x^2 + a_1x + a_0$	1.896
polynomials of degree 4	$\sum_{i=0}^{4} a_ix^i$	2.151
$\vdots$		$\vdots$
polynomials of degree n	$\sum_{i=0}^{n} a_ix^i$	$\Omega(n/\ln n)$

latency of all arcs. (Recall that in their model each path consists of a single arc.) With respect to the maximum user-latency and for instances with a single source and a single sink, the worst-case guarantee for user equilibria coincides with the bounds presented in Table 2.1. If general instances are considered, users at equilibrium may have arbitrarily high latencies. Finally, equilibria are also inefficient with respect to the maximum latency of all arcs.

Different studies have found that equilibria in realistic networks are closer to solutions with minimal travel time than predicted by worst case analysis. Qiu et al. (2003) conducted an empirical study with network topologies similar to sections of the Internet backbone. They report that equilibria and system optima are very similar and both perform much better than the traffic generated by the currently implemented Internet protocols, designed to minimize the routers' workload. However, the improvement of the total latency of system optimum and equilibrium solutions comes at the expense of increased congestion in the network's bottlenecks (Section 6.6 discusses this issue theoretically). Friedman (2003) explained why small losses should be expected in most instances by considering generic demands instead of worst-case ones. For instance, he showed that the Lebesgue measure of the set of instances with high coordination ratio is small. Mahmassani and Peeta (1993), and Wie et al. (1995) compared user equilibria to optimal flows in dynamic networks. For the small instances they analyzed, the conclusion is that equilibria are, on average, 10% more costly than optimal solutions. In

Chapter 4, we simulate instances arising from real-world traffic networks with similar findings.

2.5.1 The Case of Linear Latency Functions

Before closing this review, we give a different proof of a result shown by Roughgarden and Tardos (2002) that establishes that the inefficiency of user equilibria in networks with linear latency functions is at most 4/3. This simpler proof demonstrates the power of the variational inequality approach and helps to set the stage for Chapter 3.

Theorem 2.9 (Roughgarden and Tardos 2002). *Consider an instance with latency functions of the form $\ell_a(x_a) = q_a x_a + r_a$ with $q_a, r_a \geqslant 0$, for $a \in A$. Let f be a user equilibrium and f^* be a system optimum corresponding to the instance, respectively. Then, $C(f) \leqslant \frac{4}{3}\, C(f^*)$.*

Proof (Correa, Schulz, and Stier-Moses 2004b). Let x be a feasible flow. Using Proposition 2.7, the inequality $(x_a - f_a/2)^2 \geqslant 0$, and adding $r_a f_a$ to every term, we get that

$$C(f) \leqslant C^f(x) = \sum_{a \in A} (q_a f_a + r_a) x_a \leqslant \sum_{a \in A} (q_a x_a + r_a) x_a + \frac{1}{4} \sum_{a \in A} q_a f_a^2 \leqslant C(x) + \frac{1}{4} C(f)\,.$$

Therefore, $\frac{3}{4}\, C(f) \leqslant C(x)$ for any feasible flow x, from where $C(f) \leqslant \frac{4}{3}\, C(f^*)$ follows. □

Let us make a remark that simultaneously is a preview: Exactly the same proof works for networks with side constraints for arc-flows. In fact, one can use Proposition 3.5 in lieu of Proposition 2.7. Moreover, Corollary 3.13 further generalizes this worst-case bound of 4/3 to travel cost functions ℓ satisfying $\ell(c\,x) \geqslant c\,\ell(x)$ for $c \in [0, 1]$ (with the only restriction that they are nonnegative, nondecreasing, and lower semicontinuous). This includes, among others, some concave functions.

Chapter 3

The Price of Anarchy for Networks with Side Constraints

The subject of this chapter is the generalization of the basic model described in Section 2.1 to a more realistic context. We discuss the relevance of network models with side constraints and less restricted families of travel cost functions. Arguably, the type of side constraints that has been most often considered by previous work and in practice are capacity constraints, which provide a way to upper bound the flow on arcs. We, therefore, provide some details regarding that particular class of constraints. We present different definitions of equilibria, and analyze their coordination ratios. Applications to specific classes of latency functions are discussed in Section 3.7. While normally we assume that latencies are continuous functions, we take a separate look at lower semicontinuous travel cost functions in Section 3.8.

This chapter is based on a research article by Correa, Schulz, and Stier-Moses (2004b). A preliminary version appeared in Schulz and Stier-Moses (2003).

3.1 Introduction

The link performance functions ℓ_a relate the traffic rates x_a of traffic on the links $a \in A$ to the average travel times. To account for congestion effects, these functions are typically nonlinear, positive, and strictly increasing with flow (Patriksson 1994, p. 29). In practice, the most frequently used functions are polynomials whose degrees and coefficients are determined from real-world data through statistical methods (Patriksson 1994, p. 70).

Branston (1976) and Larsson and Patriksson (1995) argued that functions of that kind are unrealistic in the sense that the resulting travel times are finite whenever the arc flows are finite, so that the arcs are actually assumed to be able to carry arbitrarily large volumes of traffic flows; in practice, however, road links have some finite limits on traffic flows.[1] Moreover, they pointed out that travel times predicted in the overloaded range do not have a real meaning. In connection with this deficiency, Hearn (1980) noted that in the basic model described in Section 2.1, "the predicted flow on some links will be far lower or far greater than the traffic engineer knows they should be if all assumptions of the model are correct." Hearn and others, in particular Larsson and Patriksson (1994, 1995, 1999) and, most recently, Marcotte, Nguyen, and Schoeb (2004), have therefore advocated the inclusion of arc flow capacities and side constraints on the arc flows as an obvious way of improving the quality of traffic assignment models.

A frequently used way to implicitly incorporate capacities is to employ volume delay formulas that tend to infinity as the arc flow approaches the arc capacity; see, e.g., Branston (1976) for a discussion (in Section 3.4.1 we use precisely those barrier functions to justify a particular equilibrium introduced in this chapter). Boyce, Janson, and Eash (1981) have empirically found that asymptotic travel time functions yield unrealistically high travel times and devious re-routing of trips. In addition, Larsson and Patriksson (1995) criticized the inherent numerical ill conditioning of this approach. They went on to exalt the extension of the basic model by including side constraints as an interesting alternative to the use of asymmetric traffic assignment models. Such extensions are made through the development of complex travel cost functions, which, in practical applications, are difficult to calibrate. In fact, the link flow pattern found by solving a model with side constraints may also be found by solving the corresponding unconstrained problem with travel time functions adjusted by the corresponding optimal

[1] Interestingly, the widely popular link delay formula proposed by the Bureau of Public Roads includes a capacity parameter, as shown in Equation (2.1).

shadow prices. The solution of a problem with side constraints can therefore be used as a tool for guiding traffic engineers in correcting the travel time functions so as to bring the flow pattern into agreement with the anticipated results (Hearn 1980). In a related application, the introduction of capacities can be used to derive tolls for the reduction of flows on overloaded links (Hearn and Ramana 1998); see Bernstein and Smith (1994) for additional references.

It is worth mentioning that some traffic control policies give rise to link flow capacity constraints (Yang and Yagar 1994), that some of the first mathematical models of traffic assignment problems used link flow capacity constraints to model congestion effects (Charnes and Cooper 1961; Jorgensen 1963), and that several authors discussed the consequences of including capacities on existing algorithms for the uncapacitated case (Daganzo 1977a; Daganzo 1977b; Hearn 1980; Hearn and Ribera 1980; Hearn and Ribera 1981; Larsson and Patriksson 1994; Larsson and Patriksson 1995). Larsson and Patriksson (1999) have summarized and extended their earlier work to general convex side constraints on the vector of arc flows.

3.2 User Equilibria with Side Constraints

In this section, we extend the notion of a user equilibrium to networks with side constraints on the vector of arc flows. As our main motivation was the consideration of capacity constraints and they are simpler, we first extend the definition of a user equilibrium to networks with arc capacities, and then consider the general case.

We start with an extension of Wardrop's first principle to the case with capacities that was first given by Jorgensen (1963). To define the *capacity constraints*, we formally associate a nonnegative capacity c_a with each arc $a \in A$ (which may be ∞). We call c_a an arc flow x feasible if it satisfies all upper bound constraints $x_a \leqslant c_a$, for $a \in A$. A path $P \in \mathcal{P}$ is said to be *unsaturated* with respect to a given feasible flow x if and only if $x_a < c_a$ for all arcs $a \in P$. Otherwise, it is called *saturated*. The behavioral assumption that we make for the following definition is that users can only switch to unsaturated paths. Then, extending Definition 2.2, an equilibrium with capacities is a flow in which each user takes a shortest path (under the prevailing conditions) among those with residual capacity.

UE with capacities

Definition 3.1. A feasible flow f is a *user equilibrium with capacities* if no OD pair has an unsaturated path with strictly smaller cost than any path used for that pair. That is, if $f_P > 0$ for $P \in \mathcal{P}_k$, then $\ell_P(f) \leqslant \min\{\ell_Q(f) : Q \in \mathcal{P}_k \text{ unsaturated}\}$.

Maugeri (1994) considered a similar model, except he assumed that capacities are associated to paths instead of arcs. He defined an equilibrium with capacities in a similar way as in Definition 3.1, although, for him, a path is saturated if its flow matches its capacity. We prefer to continue with the model with capacities (and side constraints) on arc flows because flow on arcs can be easily measured and capacities have a physical meaning.

Let us note that the previous definition of equilibrium and the results related to it that we shall present are valid without modifications if the feasible region is defined by *generalized capacity constraints* (Larsson and Patriksson 1999). This extension models capacity constraints that may involve multiple arcs. Essentially, a generalized capacity constraint is one that satisfies that, starting from any flow, a decrease in the flow along any path cannot violate the constraint. To complete the extension of Definition 3.1, we must add that a path is saturated when it cannot accept more flow at the current congestion level.

In the unconstrained case, Definition 3.1 is obviously equivalent to Wardrop's first principle, since saturation is not an issue. In particular, all used paths in $\mathcal{P}_k$ are of equal (and minimal) latency. In contrast, in a user equilibrium with (generalized) capacities, the flow-carrying paths between the same OD pair can have different latencies (and are therefore not necessarily of minimal length). However, a user equilibrium with capacities satisfies a generalization of Wardrop's first principle.

Proposition 3.2. *Let f be a feasible flow of an instance with (generalized) capacity constraints, and continuous and nondecreasing latency functions. Then, f is an equilibrium with capacities if and only if the following property holds for all $P \in \mathcal{P}_k$. If $\ell_P(f) > L_k(f)$, then $f_P = 0$; if $\ell_P(f) < L_k(f)$, then P is saturated.*

In other words, we can partition $\mathcal{P}_k$ into three sets: paths that are short and saturated, paths that have a common length equal to $L_k(f)$, and longer paths without flow.

We should remark that our definition of a user equilibrium with capacities includes solutions that Marcotte, Nguyen, and Schoeb (2004) considered "less natural" because a group of users that travels on a long path could contribute to the saturation of a shorter path that they would prefer but cannot take. The reason is precisely that the shortest path is saturated by their own presence. To prevent this possible anomaly, we also extend the definition of a Nash equilibrium to the case with capacities. The corresponding behavioral assumption is that a user may consider switching to a certain path only if the solution satisfies the capacity constraints after the switch. Referring to those paths as feasible, we say that a flow is at equilibrium if no arbitrary small bundle of users has any incentive to switch to another feasible path. Essentially, this definition was first suggested by Bernstein and Smith (1994).

Although Bernstein and Smith, and Marcotte, Nguyen, and Schoeb only looked at capacity constraints, following the previous discussion, we generalize Definition 2.3 to the case of arbitrary side constraints. Let us denote the space of arc flows satisfying those constraints by the convex and closed set $X \subseteq \mathbb{R}^A_{\geqslant 0}$. (It is easy to see that with open sets, equilibria may fail to exist.) Furthermore, we refer to an arbitrary flow f as *feasible* when it satisfies demands ($\sum_{P\in\mathcal{P}_k} f_P = d_k$) and its projection into the space of arc flows belongs to the set X. For convenience, we henceforth consider instances that possess feasible flows.

X

Definition 3.3. A feasible flow f is a *user equilibrium with side constraints* if the following condition holds for all $k \in K$, all paths $Q, R \in \mathcal{P}_k$ such that $f_Q > 0$, and all $0 \leqslant \varepsilon \leqslant \bar{\varepsilon}$ for a small $\bar{\varepsilon}$. Namely, a new flow f^ε, defined by

UE with side constraints

$$f^\varepsilon_P \overset{\text{def}}{=} \begin{cases} f_Q - \varepsilon & \text{if } P = Q, \\ f_R + \varepsilon & \text{if } P = R, \\ f_P & \text{otherwise,} \end{cases} \qquad \text{for } P \in \mathcal{P}\,,$$

must satisfy that $\ell_Q(f) \leqslant \ell_R(f^\varepsilon)$ whenever it is feasible.

While Property 2.4 showed that Wardrop and Nash equilibria are equivalent for unconstrained networks with continuous and monotone latency functions, this is not necessarily true for their extensions to the case with constraints. For instance, the problem alluded to by Marcotte, Nguyen, and Schoeb is obviously eliminated by Definition 3.3 because users have the possibility of switching to paths saturated by themselves. In fact, although Definition 3.3 can be considered more general than 3.1 because it allows us to incorporate arbitrary side constraints, under the case of (generalized) capacity constraints, both apply and Definition 3.3 is more restrictive.

Proposition 3.4. *Consider an instance with (generalized) capacity constraints, and continuous and nondecreasing latency functions. If a feasible flow f satisfies Definition 3.3, then f satisfies Definition 3.1.*

Proof. Consider a path $Q \in \mathcal{P}_k$ such that $f_Q > 0$. We have to show that $\ell_Q(f) \leqslant \ell_R(f)$ for each unsaturated path $R \in \mathcal{P}_k$. That easily follows the continuity of the latency functions, and from Definition 3.3 by using the same Q and R in that definition. □

All the examples and results presented hereafter are valid for Definition 3.3, and therefore for the more general definition (when it applies), as the previous proposition shows. Moreover, the particular equilibrium that we single out in Section 3.4 to overcome the difficulty of characterizing the best user equilibrium with side constraints in a not necessarily convex space, satisfies Definition 3.3, too. In Section 3.8, we relax the assumption of continuity and compare the different notions of equilibria again.

Let us finally remark that Marcotte, Nguyen, and Schoeb went with a more involved definition of user equilibrium with capacities. Instead of path-based formulations as those we have just described, they considered that each user plans a strategy that ranks the possible choices at each intersection. If, at some node along the trip, the first option listed in the strategy is not available because the corresponding arc is saturated, the user will try the second option, and so on. This mechanism is a random process because if there is more demand for an arc than its capacity, a random selection will take place.

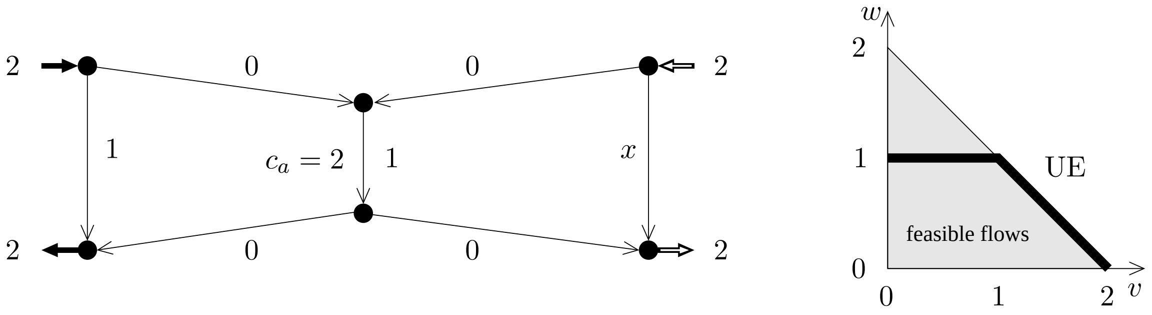


Figure 3-1: Example showing that the set of user equilibria with side constraints may be nonconvex. The instance, displayed on the left, has a single arc with a finite capacity. The graph on the right depicts the space of flows. The heavy solid line represents the set of user equilibria.

Some users will be allowed to go ahead and the rest will try their next choices. With this process in place, an equilibrium was defined as an assignment of users to strategies such that users do not have any incentive to deviate and improve their expected travel times (the travel time of a user is random).

3.3 Inefficiency, Nonuniqueness, and Nonconvexity of User Equilibria with Side Constraints

In networks without constraints, the user equilibrium is essentially unique; in particular, different equilibria, if any, share the same total latency. An important effect of side constraints is the existence of multiple equilibria, which is caused by the restrictions imposed by the side constraints. The instance shown on the left of Figure 3-1 provides an example with two commodities. The nodes on the left represent one OD pair, while the nodes on the right form the other OD pair. The demand between each consists of 2 units of flow. Arc labels indicate the corresponding latency functions and the arc in the center is the only one with finite capacity. Every user has two options: The route that goes through the center, and the alternative at the side. One can represent any feasible flow in this network using two variables.

Let v and w denote the flow that is routed through the common arc corresponding to the left and the right OD pair, respectively. Thus, the set of all feasible flows is in one-to-one correspondence with $\{v, w \in [0,2] : v + w \leqslant 2\}$ because the capacity constraint must be obeyed and the flow on the four paths must be nonnegative. This space of flows is shown on the right of Figure 3-1. According to Definition 3.1, a feasible flow is a capacitated user equilibrium if and only if at least one of the following two conditions holds:

(i) $w = 1$, i.e., the travel times along both paths for the OD pair on the right are the same;

(ii) $v+w = 2$ and $w < 1$, i.e., the common arc is used up to capacity and the alternative path for the OD pair on the right has higher cost.

Consequently, multiple equilibria with different total travel times can exist. This example additionally shows that the space of equilibria is in general not convex. Indeed, the thick black line of Figure 3-1 represents the set of all user equilibria.

Suppose that users will keep switching routes as long as they have a better choice. Then, the only stable solutions are the equilibria of the network game. As the model without side constraints has essentially a single equilibrium, no user or group of users has any incentive to deviate because they know that even if they eventually converge to another equilibrium, that equilibrium will not be better for them. In contrast to that, with side constraints, although no user has any incentive to deviate, groups of users can cooperate to improve their latencies. A group of users may switch routes so the resulting flow is also an equilibrium in such a way that all of them are better off in the new solution. In the previous example, one user of the left OD pair taking the route through the center, and another user of the right OD pair taking the route at the side, can cooperate and improve both the quality of the equilibrium and their experienced latency (as long as $w < 1$). Therefore, the model with side constraints can benefit from coordination because users could be guided towards a good equilibrium. If solutions that are not at equilibrium can be accepted by users, Chapter 4 suggests a way to achieve an approximate equilibrium that is more efficient than an exact one. Using that solution, users do not have a big incentive to deviate.

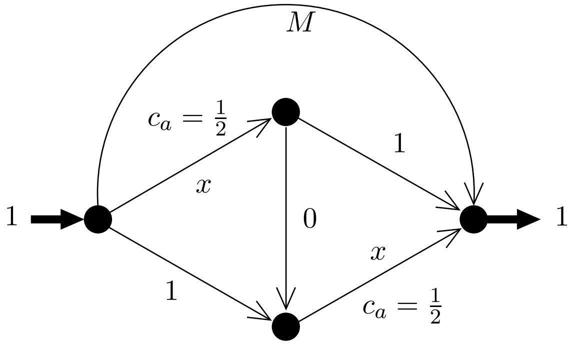


Figure 3-2: Instance with arbitrarily bad equilibria

Besides the existence of multiple equilibria, the price of anarchy for the model with capacities is, in general, unbounded. For that, consider the single commodity instance shown in Figure 3-2. Arc labels again represent the corresponding latency functions; two arcs have finite capacity. The flow that routes 1/2 on the only path consisting of three arcs and 1/2 on the arc with constant latency M is a user equilibrium with capacities. Its total travel time is $\frac{1}{2}\left(\frac{1}{2}+0+\frac{1}{2}\right)+\frac{1}{2}M=\frac{1}{2}(M+1)$. On the other hand, the system-optimal flow, which incidentally happens to be another user equilibrium with capacities, routes 1/2 on each of the two paths with two arcs. Its total travel time is $2\frac{1}{2}(\frac{1}{2}+1)=\frac{3}{2}$. Clearly, the ratio of the two values goes to infinity as $M \to \infty$.

3.4 The Beckmann User Equilibrium

Recall that Section 2.3 describes a mathematical program for computing user equilibria in the basic model that was introduced by Beckmann, McGuire, and Winsten (1956). The natural way of extending it to our more general case is the inclusion of the additional side constraints. To that effect, we define a *Beckmann user equilibrium* to be an optimal solution to the following problem: BUE

$$\min \quad \sum_{a\in A}\int_0^{f_a}\ell_a(x)\,dx \tag{3.1a}$$

$$\text{s.t.} \quad \sum_{P\ni a} f_P = f_a \qquad a\in A, \tag{3.1b}$$

$$\sum_{P \in \mathcal{P}_k} f_P = d_k \qquad k \in K, \tag{3.1c}$$

$$(f_a)_{a \in A} \in X\,. \tag{3.1d}$$

As this amounts to minimizing a convex function over a nonempty convex set, the set of optimal flows is nonempty and convex. For the example in the previous section (see Figure 3-1), the set of all Beckmann user equilibria corresponds with the set $\{0 \leqslant v \leqslant 1, w = 1\}$, which is denoted as BUE in Figure 3-3. Note that a Beckmann user equilibrium is not necessarily the most efficient equilibrium; it is just one that has a good characterization. It is this structure that helps us to carry forward some of the results known from networks without side constraints. First-order optimality conditions imply that a flow f is a Beckmann user equilibrium if and only if

$$\text{for all feasible directions } h: \sum_{a \in A} h_a \ell_a(f_a) \geqslant 0\,.$$

If we let x be any feasible flow, $x - f$ is a feasible direction at f (and all feasible directions can be obtained in this way). Therefore, the last equation is equivalent to

$$\text{for all feasible flows } x: \sum_{a \in A} (x_a - f_a) \ell_a(f_a) \geqslant 0\,. \tag{3.2}$$

Condition (3.2) can stated as a variational inequality, extending that of the basic model introduced in Proposition 2.7. Actually, we could have defined Beckmann user equilibrium as the solution to the following variational inequality problem because this is the only property we shall use in the proofs that follow. We prefer the mathematical program as it clearly shows that the problem can be solved in polynomial time and that the set of optimal solutions is nonempty and convex.

BUE condition

Proposition 3.5. *Consider an instance with side constraints, and continuous and nondecreasing latency functions. A feasible flow f is a Beckmann user equilibrium if and only if*

$$\textit{for all feasible flows } x: \; C^f(f) \leqslant C^f(x)\,. \tag{3.3}$$

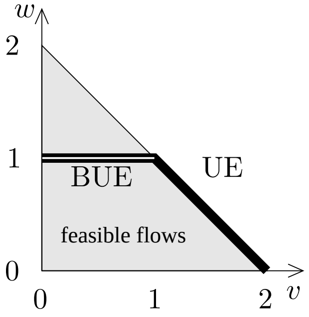


Figure 3-3: Convexity of Beckmann user equilibria, here called BUE.

Like its counterpart (2.8) for uncapacitated networks, (3.3) is crucial for proving results on the efficiency of (Beckmann) user equilibria. But, let us first show that a Beckmann user equilibrium is indeed a user equilibrium with side constraints in the sense of Definition 3.3. Note that Larsson and Patriksson (1999, Theorem 2.7) presented a similar result, although they did not define user equilibrium with side constraints as we have done in Section 3.1. Namely, they proved that (what we called) a Beckmann user equilibrium satisfies the generalized Wardrop condition given by Property 3.2. While they needed a feasible set given by generalized capacity constraints, the next lemma holds with an arbitrary feasibility set.

Lemma 3.6. *Consider an instance with side constraints, and continuous and nondecreasing latency functions. If a feasible flow f is a Beckmann user equilibrium, then f is a user equilibrium with side constraints.*

Proof. To show that f satisfies Definition 3.3, suppose to the contrary that there are two paths $Q, R \in \mathcal{P}_k$ for some OD pair k with $f_Q > 0$ such that $\ell_R(f^\varepsilon) < \ell_Q(f)$, where

$$f^\varepsilon_P = \begin{cases} f_Q - \varepsilon & \text{if } P = Q, \\ f_R + \varepsilon & \text{if } P = R, \\ f_P & \text{otherwise,} \end{cases} \qquad \text{for } P \in \mathcal{P}$$

is a feasible flow for all $0 < \varepsilon \leqslant \overline{\varepsilon}$ for some $\overline{\varepsilon}$. Now, keep $f^{\overline{\varepsilon}}$ fixed and consider

$$\sum_{a \in A} (f_a^{\overline{\varepsilon}} - f_a)\ell_a(f_a) = \sum_{P \in \mathcal{P}} (f_P^{\overline{\varepsilon}} - f_P)\ell_P(f) = \overline{\varepsilon}\left(\ell_R(f) - \ell_Q(f)\right).$$

Since latency functions are continuous and nondecreasing, it follows that $\ell_R(f) - \ell_Q(f) < 0$ and so we have a contradiction to (3.2). □

Although we do not need to assume that latency functions are continuous to prove the existence of a Beckmann user equilibrium, the continuity assumption was important for the previous lemma. Indeed, with arbitrary latencies, a user equilibrium with side constraints may fail to exist or a Beckmann user equilibrium may not be an equilibrium with side constraints. In Section 3.8, we present an example that illustrates this situation and further discuss the issue of discontinuous latencies.

As a historical note, user equilibria in networks with capacities, and more generally with side constraints, have been considered before (see Jorgensen 1963; Daganzo 1977a; Daganzo 1977b; Hearn 1980; Hearn and Ribera 1980; Hearn and Ribera 1981; Maugeri 1994; Larsson and Patriksson 1995; Larsson and Patriksson 1999, as well as Patriksson 1994 and the references therein). With the exception of the articles by Jorgensen and Maugeri, the mentioned articles defined a user equilibrium with capacities as our notion of Beckmann user equilibrium. We believe that this is not the correct definition of user equilibrium because it is given by an optimization problem and not by a description of user behavior. We do not know of any empirical study that establishes that, in practice, the prevailing condition is close to a Beckmann user equilibrium. However, one of the contributions of the next section is providing some theoretical evidence supporting that claim.

3.4.1 Further Remarks for Networks with Capacities

Hearn (1980) noted that a Beckmann user equilibrium of a network with capacities is an uncapacitated user equilibrium with respect to latencies $\ell_a(\cdot) + \gamma_a$, where $\gamma_a \in \mathbb{R}_{\geqslant 0}$ is the shadow price (Karush-Kuhn-Tucker multiplier) of the capacity constraint $x_a \leqslant c_a$ for arc $a \in A$ in an optimal solution to problem (3.1a) – (3.1d). Those shadow prices were interpreted by some authors as the tolls that users are willing to pay (Jorgensen 1963) or queuing delays that users experience in addition to the time to traverse the links (Payne and Thompson 1975; Inouye 1987; Nesterov 2000; Nesterov and de Palma 2000).

This point of view facilitates an alternative proof of Proposition 3.5. In fact, let f be a Beckmann user equilibrium and x be any feasible flow. Then we can re-prove (3.3) as follows:

$$\begin{aligned}
C^f(f) &= \sum_{a \in A} \ell_a(f_a) f_a + \sum_{a \in A} \gamma_a (f_a - c_a) \\
&= \sum_{a \in A} (\ell_a(f_a) + \gamma_a) f_a - \sum_{a \in A} \gamma_a c_a \\
&\leqslant \sum_{a \in A} (\ell_a(f_a) + \gamma_a) x_a - \sum_{a \in A} \gamma_a c_a \\
&= \sum_{a \in A} \ell_a(f_a) x_a + \sum_{a \in A} \gamma_a (x_a - c_a) \\
&\leqslant C^f(x)\,.
\end{aligned}$$

Here, the first equality follows from complementary slackness. The first inequality uses Proposition 2.7 for uncapacitated user equilibria, while the second one makes use of the feasibility of x.

Let us now discuss another good reason for paying attention to Beckmann user equilibria. Suppose one would forgo explicit arc capacities and would instead incorporate barrier terms in the latency functions. More specifically, let $\mu \in \mathbb{R}_{\geqslant 0}$ be a penalty parameter and consider the modified latency functions $\ell_a^\mu(x_a) \overset{\text{def}}{=} \ell_a(x_a) + \mu/(c_a - x_a)$ for all arcs a with finite capacities, with the understanding that the barrier term equals $+\infty$ for $x_a \geqslant c_a$. The next lemma essentially shows that in the limit (for $\mu \to 0$), selfish users behave like they would in a Beckmann user equilibrium.

Lemma 3.7. *Let (μ_i) be a parameter sequence with $0 < \mu_{i+1} < \mu_i$ for $i \in \mathbb{N}$, and $\mu_i \to 0$. Let (f^i) be the corresponding sequence of user equilibria in the network without capacities but with modified latencies $\ell_a^{\mu_i}$. Every limit point of the sequence (f^i) is a*

Beckmann user equilibrium of the original instance (i.e., with capacities).[2]

Proof. According to Beckmann, McGuire, and Winsten (1956), each user equilibrium f^i minimizes the following objective function, subject to (3.1b), (3.1c), and (3.1d):

$$\sum_{a \in A} \int_0^{f_a} \left(\ell_a(x) + \frac{\mu_i}{c_a - x} \right) dx \; . \tag{3.4}$$

Hence, f^i also minimizes

$$\sum_{a \in A} \int_0^{f_a} \ell_a(x)\, dx \; - \; \mu_i \sum_{a \in A} \ln(c_a - f_a) \; , \tag{3.5}$$

which differs from (3.4) by a constant. As the second term in (3.5) is a barrier function as well, it follows that each limit point of (f^i) is an optimal solution of the original problem (3.1a) – (3.1d) (see, e.g., Bertsekas 1999, Proposition 4.1.1). □

In addition, we can prove a similar result for the sequence of system optima. The proof follows the same idea as that of the previous lemma.

Lemma 3.8. *Let (μ_i) be a parameter sequence with $0 < \mu_{i+1} < \mu_i$ for $i \in \mathbb{N}$, and $\mu_i \to 0$. Let $(f^{i,*})$ be the corresponding sequence of system optima in the network without capacities but with modified latencies $\ell_a^{\mu_i}$. Every limit point of the sequence $(f^{i,*})$ is a system optimum of the original instance.*

In spite of the last two results, it is not true that the coordination ratio of an instance in which capacities are enforced by using modified latency functions approaches the coordination ratio of a Beckmann user equilibrium of the capacitated instance. In other words, if the subsequence (f^i) of user equilibria converges to the Beckmann user equilibrium f, then in general

$$\frac{C^{\mu_i}(f^i)}{C^{\mu_i}(f^{i,*})} \quad \overset{\mu_i \to 0}{\not\to} \quad \frac{C(f)}{C(f^*)} \, . \tag{3.6}$$

[2] The sequence can be shown to converge if there is a single Beckmann user equilibrium.

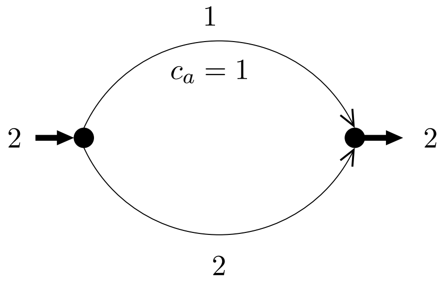


Figure 3-4: The coordination ratio with barriers does not converge to that with capacities

Here, f^* and $f^{i,*}$ are system-optimal solutions corresponding to the instances for which f and f^i are user equilibria, respectively.

For an example, consider a network of two parallel arcs connecting a single origin with a single destination, and a demand rate of 2, as shown in Figure 3-4. One of the arcs has unit latency and unit capacity while the second arc has latency equal to 2 and infinite capacity. Both the user equilibrium with capacities and the system optimum route one unit of flow along each arc. The total travel time of both solutions is 3. If we try to enforce the capacity constraint of the first arc with the help of a barrier term, its latency becomes $1+\mu(1-x)^{-1}$. The corresponding user equilibrium is $(1-\mu, 1+\mu)$; here, the first coordinate refers to the capacitated arc. As both latency functions evaluate to 2, the total travel time is 4. The optimal flow is $(1-\sqrt{\mu}, 1+\sqrt{\mu})$, and its total travel time is $3-\mu+2\sqrt{\mu}$. While the sequence $\{(1-\mu, 1+\mu)\}$ converges to the capacitated user equilibrium $(1,1)$ for $\mu \to 0$, the corresponding sequence of total travel times remains constant at 4. Hence, the left-hand side of (3.6) converges to 4/3 and not to 1, the value of the right-hand side.

3.5 The Efficiency of Beckmann User Equilibria

We now present upper bounds on the inefficiency of any Beckmann equilibrium. Recall from Section 3.3 that an arbitrary user equilibrium with side constraints can be arbitrarily inefficient (in contrast to the situation in networks without constraints). We first focus on a bicriteria result *à la* Roughgarden and Tardos (2002, Theorem 3.1). This result implies that an equilibrium is not worse than the system optimum of an instance with twice the demand. Put differently, lacking coordination is not more expensive than doubling the demand.

Theorem 3.9. *Consider an instance of the traffic assignment model with side constraints, and continuous and nondecreasing latency functions. If f is a Beckmann user equilibrium for that instance and x is a feasible flow, then $C(f) \leqslant C(2x)$, where $2x$ denotes a solution that routes twice as much flow along every arc.*

Proof. Like Roughgarden and Tardos, we start by modifying the original latency functions ℓ_a. Namely,

$$\tilde{\ell}_a(x_a) \overset{\text{def}}{=} \begin{cases} \ell_a(f_a) & \text{if } x_a \leqslant f_a, \\ \ell_a(x_a) & \text{if } x_a \geqslant f_a. \end{cases}$$

Consider a flow y with arbitrary total value (i.e, possibly infeasible). The increase in the cost of y with respect to the new latencies is bounded by the following expression:

$$\widetilde{C}(y) - C(y) = \sum_{a\in A}(\tilde{\ell}_a(y_a) - \ell_a(y_a))y_a \leqslant \sum_{a\in A}\ell_a(f_a)f_a = C(f)\,,$$

where the inequality follows directly from the definition of $\tilde{\ell}$. Using $\tilde{\ell}_P(y) \geqslant \tilde{\ell}_P(0) = \ell_P(f)$ for any path P, we also obtain:

$$\widetilde{C}(y) = \sum_{P\in\mathcal{P}}\tilde{\ell}_P(y)y_P \geqslant \sum_{P\in\mathcal{P}}\ell_P(f)y_P = C^f(y)\,.$$

Finally, for a feasible flow x, Condition (3.3) implies that $C(f) \leqslant C^f(x)$. Putting the three inequalities together yields

$$C(f) = 2\,C(f) - C(f) \leqslant 2\,C^f(x) - C(f) = C^f(2x) - C(f) \leqslant \widetilde{C}(2x) - C(f) \leqslant C(2x)\,.$$

□

It is straightforward to generalize the previous theorem using different coefficients. Namely, following Roughgarden, for any constant $\eta > 1$ and any feasible flow x, $C(f) \leqslant (\eta - 1)^{-1}\,C(\eta x)\,$.

We now turn our attention to the inefficiency of the best equilibrium with side constraints. Note that the early paper of Jorgensen (1963) proved that system optima are at equilibrium when latencies are constant functions. Our main result bounds the inefficiency of a Beckmann user equilibrium when latencies are drawn from a given set $\mathcal{L}$. We shall continue to assume that latency functions are just continuous and nondecreasing. For example, $\mathcal{L}$ could be the polynomials of degree at most n. For every function $\ell \in \mathcal{L}$ and every value $v \geqslant 0$, let us define:

$$\beta(v,\ell) \overset{\text{def}}{=} \frac{1}{v\,\ell(v)} \max_{x \geqslant 0} \left\{ x\big(\ell(v) - \ell(x)\big) \right\}, \tag{3.7}$$ $\beta(v,\ell)$

where by convention $0/0 = 0$. It is obvious that $\beta(v,\ell) \geqslant 0$ and since $x(\ell(v) - \ell(x)) \leqslant 0$ for $x > v$, we could have restricted the maximum to the interval $[0, v]$. In addition, let us define $\beta(\ell) \overset{\text{def}}{=} \sup_{v \geqslant 0} \beta(v,\ell)$ and $\beta(\mathcal{L}) \overset{\text{def}}{=} \sup_{\ell \in \mathcal{L}} \beta(\ell)$. Note that $\beta(\mathcal{L}) \leqslant 1$ because the maximum in (3.7) cannot be larger than $v\ell(v)$. These definitions arise naturally from the result we are about to present, which is a straightforward extension of Theorem 2.9. $\beta(\ell)$ $\beta(\mathcal{L})$

Theorem 3.10. *Let $\mathcal{L}$ be a family of continuous and nondecreasing latency functions. Consider an instance of the side-constrained traffic assignment model* (3.1b) – (3.1d) *with latency functions drawn from $\mathcal{L}$. Then, the ratio of the total travel time of a Beckmann user equilibrium f to that of a system optimum f^* is bounded from above by $(1 - \beta(\mathcal{L}))^{-1}$, i.e.,*

$$C(f) \leqslant \frac{1}{1 - \beta(\mathcal{L})}\, C(f^*)\,.$$

Proof. Let x be a feasible flow. By definition $C^f(x) = \sum_{a \in A} \ell_a(f_a) x_a$; hence,

$$C^f(x) \leqslant \sum_{a \in A} \beta(f_a, \ell_a) \ell_a(f_a) f_a + \sum_{a \in A} \ell_a(x_a) x_a \leqslant \beta(\mathcal{L}) C(f) + C(x)\,. \tag{3.8}$$

From Proposition 3.5, $C(f) \leqslant C^f(x)$, and the claim follows by applying (3.8) to $x = f^*$ □

In spite of the simplicity of its proof, the power and flexibility of the last theorem will become evident when we relate it to the main result of Roughgarden (2003b) next and demonstrate several further implications in Section 3.7. The key was to get the definition of $\beta(\mathcal{L})$ "right."

3.6 Comparison with Previous Results

Let $\mathcal{L}$ be a given family of latency functions. We now relate $\beta(\mathcal{L})$ to the anarchy value $\alpha(\mathcal{L})$ introduced by Roughgarden (2003b). In order to do so, we have to assume that, in addition to being continuous and monotone, ℓ is differentiable and s-convex for all $\ell \in \mathcal{L}$ (the setting of Roughgarden). The *anarchy value* $\alpha(\ell)$ of a latency function ℓ is

$\alpha(\ell)$

$$\alpha(\ell) \overset{\text{def}}{=} \sup_{v>0:\, \ell(v)>0} \left[\lambda \frac{\ell(\lambda v)}{\ell(v)} + (1-\lambda)\right]^{-1},$$

where $\lambda \in [0,1]$ solves $\ell^*(\lambda v) = \ell(v)$. By rearranging terms,

$$\alpha(\ell) = \left[1 - \sup_{v>0:\, \ell(v)>0} \lambda \left(\frac{\ell(v) - \ell(\lambda v)}{\ell(v)}\right)\right]^{-1},$$

we can prove that $\alpha(\ell) = (1-\beta(\ell))^{-1}$. Indeed, if we use $x = \lambda v$ in the definition of $\beta(\ell)$, it is clear that $\alpha(\ell) \leqslant (1-\beta(\ell))^{-1}$. For the other inequality, consider a given v. Since $x(\ell(v) - \ell(x))$ is concave and its value in 0 and v is zero, there is a point $x^* \in (0, v)$ that attains the maximum. From the differentiability of ℓ, $(x(\ell(v) - \ell(x)))'$ evaluated at $x = x^*$ equals zero. Therefore, $\lambda = x^*/v$ satisfies $\ell^*(\lambda v) = \ell(v)$, as required.

$\alpha(\mathcal{L})$ Hence, the anarchy value $\alpha(\mathcal{L}) \overset{\text{def}}{=} \sup_{\ell \in \mathcal{L}} \alpha(\ell)$ of a class $\mathcal{L}$ is equal to $(1-\beta(\mathcal{L}))^{-1}$. Therefore, Theorem 3.10 not only implies Roughgarden's main result (Roughgarden 2003b, Theorem 3.8) but also extends it to functions ℓ that are not necessarily differentiable or s-convex. Moreover, the constraints imposed on arc flows do not matter.

We conclude this section by showing that the bound given in Theorem 3.10 is tight. In fact, if $\mathcal{L}$ contains the constant functions, this bound is attained by a single-commodity network consisting of two parallel arcs, which essentially reflects the *independence of the network topology* property highlighted by Roughgarden. Let us assume that the value $\beta(\mathcal{L})$ is achieved for $\ell \in \mathcal{L}$ and $v > 0$. (Although we could use a convergent sequence

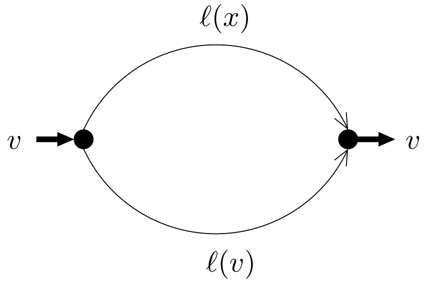


Figure 3-5: Generalization of Pigou's example

if the supremum is not attained, we omit this analysis since it is easy and does not provide further insights.) Consider the network depicted in Figure 3-5 (Pigou 1920; Roughgarden 2003b), with two parallel links, one with latency $\ell(x)$ and the other with constant latency $\ell(v)$. A demand of v is to be routed. In this situation, the cost of the equilibrium f is $C(f) = v\,\ell(v)$, while the system optimum f^* can be evaluated as follows:

$$C(f^*) = \min_{0\leqslant x\leqslant v} \{x\,\ell(x) + \ell(v)(v-x)\} = v\,\ell(v) - \max_{0\leqslant x\leqslant v} \{x\,(\ell(v) - \ell(x))\}\,.$$

Hence, the ratio between the total latency of the user equilibrium and that of the system optimum is

$$\frac{C(f)}{C(f^*)} = \left(1 - \frac{\max_{x\geqslant 0} \{x(\ell(v) - \ell(x))\}}{v\,\ell(v)}\right)^{-1} = (1-\beta(\mathcal{L}))^{-1}\,.$$

3.7 Computing the Price of Anarchy

Since the results in the last subsection generalize the results by Roughgarden (2003b), the bounds he obtains for specific classes of latencies (e.g., linear functions and polynomials with positive coefficients) apply here as well. In this section we study bounds for more general latency functions. We start with two auxiliary lemmas.

Lemma 3.11. *Let $\mathcal{L}$ be a family of continuous and nondecreasing latency functions. Assume that for some real function s, ℓ satisfies $\ell(c\,x) \geqslant s(c)\ell(x)$ for all $c \in [0,1]$.*

Then,

$$\beta(\mathcal{L}) \leqslant \sup_{0\leqslant x\leqslant 1} \{x(1-s(x))\}\,.$$

Proof. Recall from (3.7) that $\beta(v,\ell)$ is defined as

$$\beta(v,\ell) = \max_{0\leqslant x\leqslant v} \left\{ \frac{x}{v}\left(1-\frac{\ell(x)}{\ell(v)}\right)\right\}\,.$$

Rewriting x as $v\,(x/v)$ and using the assumption, we can bound this expression from above by

$$\sup_{0\leqslant x\leqslant v} \left\{ \frac{x}{v}\left(1-s\left(\frac{x}{v}\right)\right)\right\} = \sup_{0\leqslant x\leqslant 1} \{x\,(1-s(x))\}\ ,$$

which implies the claim. □

Lemma 3.12. *Let $\mathcal{L}$ be a family of continuous and nondecreasing latency functions Assume that for some real function s, ℓ satisfies $\ell(c\,x) \geqslant s(c)+\ell(x)$ for all $c \in [0,1]$. Then,*

$$C(f) \leqslant C(f^*) - |A|d \inf_{0\leqslant x\leqslant 1}\{x\,s(x)\}\,,$$

where $d = \sum_{k\in K} d_k$ is the total demand to be routed, f is a Beckmann user equilibrium, and f^* is a system optimum.

Proof. In this case, it is easy to see that

$$\ell(v)\beta(v,\ell) \leqslant \sup_{0\leqslant x\leqslant v}\left\{-\frac{x}{v}\ s\left(\frac{x}{v}\right)\right\} = -\inf_{0\leqslant x\leqslant 1}\{x\,s(x)\}\,.$$

If we plug this into (3.8) with $\ell=\ell_a$ and $v=f_a$, we obtain,

$$C(f) \leqslant \sum_{a\in A}\beta_a(f_a,\ell_a)\ell_a(f_a)f_a + \sum_{a\in A}\ell_a(f_a^*)f_a^* \leqslant C(f^*) - |A|d \inf_{0\leqslant x\leqslant 1}\{x\,s(x)\}\,.$$

□

We now apply Lemmas 3.11 and 3.12 to specific classes of latency functions. The following corollaries extend Theorem 2.9. Indeed, the Corollary 3.13 implies that the price of anarchy is 4/3 for all nonnegative concave functions and this bound still holds in networks with side constraints (assuming that a Beckmann user equilibrium is chosen). Corollary 3.14 generalizes Roughgarden's bound for polynomials of degree n with positive coefficients.

Corollary 3.13. *If the set $\mathcal{L}$ of continuous and nondecreasing latency functions is contained in the set $\{\ell : \ell(c\,x) \geqslant c\,\ell(x) \text{ for } c \in [0,1]\}$, then $(1-\beta(\mathcal{L}))^{-1} \leqslant 4/3$.*

Proof. Use Lemma 3.11 and note that $\sup\limits_{0\leqslant x\leqslant 1} \{x(1-x)\} = \frac{1}{4}$. □

Corollary 3.14. *If the set $\mathcal{L}$ of continuous and nondecreasing latency functions is contained in the set $\{\ell : \ell(c\,x) \geqslant c^n\,\ell(x) \text{ for } c \in [0,1]\}$ for some positive number n, then*

$$(1-\beta(\mathcal{L}))^{-1} \leqslant \frac{(n+1)^{1+1/n}}{(n+1)^{1+1/n}-n}\,.$$

Proof. Use Lemma 3.11 and note that $\sup\limits_{0\leqslant x\leqslant 1} \{x(1-x^n)\} = \dfrac{n}{(n+1)^{1+1/n}}$. □

In particular, the price of anarchy in networks with quadratic (resp. cubic) latency functions is 1.626 (resp. 1.896). Table 2.1 lists more values, which were computed using the formula we just derived.

Finally, the following result comprises the case in which latency functions are logarithmic (i.e., $\ell(x) = \log(1+x)$). The Beckmann user equilibrium offers an additive performance guarantee in this situation.

Corollary 3.15. *If the set $\mathcal{L}$ of continuous and nondecreasing latency functions is contained in the set $\{\ell(\cdot) : \ell(c\,x) \geqslant \log_b(c) + \ell(x) \text{ for } c \in [0,1]\}$, then*

$$C(f) \leqslant C(f^*) + \frac{|A|d}{e\ln b}\,.$$

Proof. Use Lemma 3.12 and note that $\inf\limits_{0\leqslant x\leqslant 1} \{x\log_b(x)\} = -\dfrac{1}{e\ln b}$. □

3.8 Lower Semicontinuous Latency Functions

Traffic assignment models customarily depend on the assumption of continuous travel cost functions. However, Bernstein and Smith (1994) have pointed out that there are

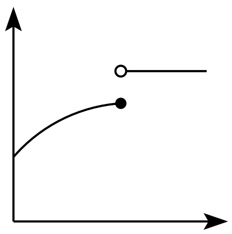


Figure 3-6: A lower semicontinuous function

times when this assumption is not appropriate. In this situation, a more careful distinction between different versions of the equilibrium concept is essential. For unconstrained networks, Section 2.5 points out that the notions of Wardrop and Nash equilibria are equivalent to each other for continuous latency functions. Let us borrow the following example from Florian and Hearn (1995) to illustrate the consequences of relaxing that assumption. Like in Figure 3-5, two parallel arcs connect an OD pair with demand rate 2. The travel cost function for the first arc is $\ell_a(x_a) = x_a$; for the second arc, it is

$$\ell_b(x_b) = \begin{cases} x_b & \text{if } x_b < 1, \\ x_b + 1 & \text{if } x_b \geqslant 1. \end{cases}$$

Although the solution $x_a = x_b = 1$ is a Beckmann user equilibrium, no solution satisfies Definitions 3.1 and 3.3, which implies that the instance does not have Wardrop or Nash equilibria.

lsc Let us recall that a real function ℓ is *lower semicontinuous* if $\ell(x) \leqslant \liminf \ell(x_n)$ for all x in its domain and all sequences (x_n) with $\lim_{n\to\infty} x_n = x$. Here, $\liminf \ell(x_n) = \lim_{n\to\infty} \inf\{\ell(x_m) : m \geqslant n\}$. In fact, if ℓ is nondecreasing and lower semicontinuous, then $\ell(x) = \lim_{y\nearrow x} \ell(y)$, and the limit always exists.[3] Figure 3-6 shows an example of such a function. Upper semicontinuity is defined similarly.

The article of de Palma and Nesterov (1998), which considers unconstrained networks, presents additional examples as well as conditions that guarantee the existence of the different notions of equilibrium. For instance, the authors illustrate that a Wardrop

[3] Recall that $\lim_{y\nearrow x} \ell(y)$ represents the limit of $(\ell(y_i))_{i\in\mathbb{N}}$ with respect to any increasing sequence $(y_i)_{i\in\mathbb{N}}$ that converges to x from below; $\lim_{y\searrow x} \ell(y)$ is similarly defined.

equilibrium might not be a Nash equilibrium (and a Nash equilibrium might not exist), a Nash equilibrium might not be a Wardrop equilibrium (and a Wardrop equilibrium might not exist), or neither of the two might exist. Furthermore, for upper semicontinuous latencies, they prove that a Nash equilibrium is both a Wardrop and a Beckmann equilibrium. More interestingly to us, they show that Beckmann user equilibria are Nash equilibria if latencies are lower semicontinuous. This result, which is similar to Lemma 3.6, provides us with a general assumption on latency functions for the existence of Nash equilibria. We present an extension to the case of arbitrary side constraints.

Proposition 3.16. *Consider an instance with side constraints, and lower semicontinuous and nondecreasing latency functions. If a feasible flow f is a Beckmann user equilibrium of that instance, then f is a user equilibrium with side constraints.*

Proof. We show that f satisfies Definition 3.3. Consider two paths $Q, R \in \mathcal{P}_k$ for some OD pair k with $f_Q > 0$, such that

$$f_P^\varepsilon = \begin{cases} f_Q - \varepsilon & \text{if } P = Q, \\ f_R + \varepsilon & \text{if } P = R, \\ f_P & \text{otherwise}, \end{cases} \qquad \text{for } P \in \mathcal{P}$$

is a feasible flow for all $0 < \varepsilon \leqslant \bar{\varepsilon}$ and some $\bar{\varepsilon}$. Now, keep $x \overset{\text{def}}{=} f^{\bar{\varepsilon}}$ fixed and consider

$$\sum_{a \in A} (x_a - f_a)(\nabla_f)_a = \sum_{P \in \mathcal{P}} (x_P - f_P)(\nabla_f)_P = \bar{\varepsilon} \left((\nabla_f)_R(f) - (\nabla_f)_Q(f) \right),$$

where ∇_f is a subgradient of $\sum_{a \in A} \int_0^{x_a} \ell_a(y)\, dy$ at f. The fact that f is a Beckmann user equilibrium, the previous equation, and lower semicontinuity imply that

$$\sum_{a \in Q \setminus R} \ell_a(f_a) \leqslant \sum_{a \in R \setminus Q} \ell_a(f_a^+),$$

which means that no arbitrarily small bundle of users can reduce their cost by switching from Q to R. □

We will now sketch that, under minor modifications, Theorem 3.10 still holds in the more general setting of latency functions that are just lower semicontinuous. (Note that we maintain the monotonicity assumption.) Hence, in this more general setting, we still have a bound on the inefficiency of the best user equilibrium with side constraints. Bernstein and Smith, as well as de Palma and Nesterov underline the importance of this class of travel cost functions.

For a feasible (arc) flow x, we redefine $C^f(x)$ to be the standard inner product between ∇_f and x, i.e., $C^f(x) \overset{\text{def}}{=} \langle \nabla_f, x \rangle$, where ∇_f is a subgradient of $\sum_{a \in A} \int_0^{x_a} \ell_a(y)\, dy$ at f satisfying the optimality conditions for (3.1a) – (3.1d). In other words, ∇_f satisfies a condition similar to (3.3), namely $\langle \nabla_f, x - f \rangle \geqslant 0$ for all feasible flows x. Moreover, note that $\lim_{y \nearrow f_a} \ell_a(y) \leqslant (\nabla_f)_a \leqslant \lim_{y \searrow f_a} \ell_a(y)$, for all $a \in A$. The first of these inequalities together with the lower semicontinuity of ℓ implies that $C(f) \leqslant C^f(f)$.

To proceed as we did in the proof of Theorem 3.10, we also need a slight technical change in the definition of $\beta(v, \ell)$, which should now be defined as

$$\beta(v, \ell) \overset{\text{def}}{=} \frac{1}{v\, \ell(v)} \max_{x \geqslant 0} \{x\, (\ell(v^+) - \ell(x))\}.$$

Here, $\ell(v^+) = \lim_{y \searrow v} \ell(y)$. After these preparations, we can complete the proof. Let x be a feasible flow. We derive

$$C^f(x) = \langle \nabla_f, x \rangle \leqslant \sum_{a \in A} \ell_a(f_a^+) x_a \leqslant \sum_{a \in A} \beta(f_a, \ell_a) \ell_a(f_a) f_a + \sum_{a \in A} \ell_a(x_a) x_a \leqslant \beta(\mathcal{L}) C(f) + C(x)\,.$$

Recall that $C(f) \leqslant C^f(f)$ by lower semicontinuity and $C^f(f) \leqslant C^f(x)$ from the optimality conditions. Therefore, the claim follows by replacing x with a system optimum f^*.

Let us further note that it appears difficult to extend our main result to families of latency functions that are not lower semicontinuous. Consider an instance consisting of two nodes connected by arcs a and b (similar to the one depicted in Figure 3-5) with unit demand. Let the latencies be $\ell_a(f_a) = 1$ and

$$\ell_b(f_b) = \begin{cases} \frac{1}{2} & \text{if } 0 \leqslant f_b < \frac{1}{2}, \\ \frac{2}{3} f_b + \frac{1}{3} & \text{if } \frac{1}{2} \leqslant f_b < 1, \\ \frac{4}{3} & \text{if } f_b \geqslant 1. \end{cases}$$

The Beckmann user equilibrium f routes all demand along arc b for a total cost of 4/3. Although a system optimum cannot be attained, it can be approximated by a flow that routes $1/2 + \varepsilon$ along a and the rest along b. For $\varepsilon \to 0$, the total cost goes to 3/4. Since our previous definition of $\beta(\mathcal{L})$ assumes that latencies are lower semicontinuous, let us consider a more pessimistic notion, for which we can still show that an analog of Theorem 3.10 does not hold. So let $\beta(v, \ell) \overset{\text{def}}{=} \frac{1}{v\,\ell(v^-)} \sup_{x \geqslant 0} \{x\,(\ell(v^+) - \ell(x^-))\}$, where $\ell(x^-) = \lim_{y \nearrow x} \ell(y)$. In the example, $\beta(\mathcal{L}) = \sup_{\ell,v} \beta(v, \ell) = 5/12$. Hence, $(1 - \beta(\mathcal{L}))^{-1} C(f^*) = \frac{12}{7} \frac{3}{4} = \frac{9}{7} < \frac{4}{3} = C(f)$. Consequently, Theorem 3.10 (or reasonable extensions thereof) does not hold for discontinuous functions in general.

3.9 Discussion

While Wardrop (1952) had used the concept of user equilibrium to *describe* user behavior in traffic networks, equilibria have been exploited in traffic management systems to *predict* and in proposals for route guidance systems to *prescribe* user behavior (e.g., Prager 1954; Steenbrink 1974; Gartner et al. 1980; Boyce 1989). Yet, Nash equilibria in generic games and user equilibria in particular are known to be inefficient, and many experts have favored in principle the difficult-to-implement system optimum (Merchant and Nemhauser 1978; Henry, Charbonnier, and Farges 1991), which guarantees that the total travel time is minimal. The results presented in this chapter provide an a posteriori justification for employing user equilibria in traffic assignment models. We have shown for a broader class of network models than considered before that the expense of working with user equilibria instead of system optima is limited. With more generality, as the base of our proofs was Condition (3.3), our results hold for any model with separable costs for which the equilibria can be characterized by a variational inequality. In particular, we have not made use of the network structure in the proofs related to the price of anarchy.

Subsequent to our work, Karakostas and Kolliopoulos (2003) considered a model with side constraints on the path flows, based on that of Maugeri (1994). The authors defined an equilibrium with side constraints as the variational inequality (3.3) and, therefore, their definition coincides with our Beckmann user equilibrium. Using different techniques from those of this present work, they proved that the price of anarchy is 4/3 for the case of linear separable latency functions and general side constraints for the flow on paths. In actual fact, while we have confined the above presentation to side constraints on the vector of arc flows, virtually all of our results apply to the more general case of convex sets of path flows ($X \subseteq \mathbb{R}^P$). The proofs would not change because the projection of such a set into the space of arc flows is convex too and, therefore, the first-order optimality conditions are still valid. As we mentioned before, we have chosen to define constraints with respect to arc flows because, as they can be observed, they are more relevant for practical applications. Moreover, bear in mind that in the more general case of path flows, computing a Beckmann user equilibrium (or equivalently, solving the variational inequality (3.3)) cannot be done in polynomial time.

Game-theoretic concepts offer an attractive way of computing approximate solutions to certain hard problems. Although our model with side constraints may have multiple equilibria and computing the best equilibrium is difficult, Theorem 3.10 implies that a provably good user equilibrium can be computed in polynomial time. Subsequent to our work, Anshelevich et al. (2003) approximated optimal solutions to a network design problem that is NP-hard with the help of Nash and approximate Nash equilibria. A related idea was used by Fotakis et al. (2002), Feldmann et al. (2003a), Lücking et al. (2003), Gairing et al. (2003), and Gairing et al. (2004) to show that it is hard to find the best and worst equilibrium of the telecommunication game of Koutsoupias and Papadimitriou, described in Section 2.5. In addition, they showed that there exist approximation algorithms for computing a Nash equilibrium with minimal or maximal social cost.

Let us finally remark that Roughgarden and Tardos (2004) generalized their traffic model to *nonatomic congestion games*, for which all of their results still hold. This class of games, first defined by Rosenthal (1973), generalizes traffic games by not considering a network at all. Users are divided into player types (corresponding to the OD pairs), and there is a set of elements (corresponding to the arcs) with cost functions that map the congestion of the element to its cost (they correspond to the latency functions). For each player type there is a strategy set, and each strategy is a subset of the basic elements (corresponding to the paths). Finally, for each strategy and element, there is a consumption rate, which represents how much of the element is used by players that select the strategy (in our traffic game, all those rates are assumed to be one).

The results of this chapter can also be extended to these more general games in a straightforward way. Consequently, their findings also hold when side constraints are present (e.g., the elements of the ground set have capacities) and the cost functions satisfy the weaker assumptions made in this chapter. Chau and Sim (2003) extended the results concerning nonatomic congestion games by Roughgarden and Tardos to *non-separable* and *symmetric* affine latencies. Non-separable means that latency functions depend on the full vector of arc-flows; symmetric refers to the fact that the Jacobian $\nabla\ell(x)$ is symmetric (for details regarding these assumptions, see, for example, Magnanti 1984). Under these assumptions, they showed that the price of anarchy is still 4/3 in the affine case, and a generalization of $\alpha(\mathcal{L})$ in the general case. Furthermore, they incorporated elastic demands which is a common feature when the demand corresponding to an OD pair is not fixed but is a function of the minimum latency between the pair. Although equilibria do not have the same performance guarantee as before, they derived a weaker general bound. Independently, Schulz and Stier-Moses (2004) also considered elastic demands but under a different model. With their model, the price of anarchy is equal to that in the inelastic case. Let us add that Johari (2004), considering elastic demands as well, proved that the price of anarchy can be arbitrary bad for his and Tsitsiklis' traffic model (we briefly describe that model in Chapter 7).

As the assumption that latencies are symmetric is very restrictive for many applications, Perakis (2004) recently generalized the network model to the case of non-separable and *asymmetric* latency functions and fixed demand. For affine latency functions, the price of anarchy increases with the degree of asymmetry of the function. In the general nonlinear case, the mentioned bound is modified to account for changes in the Jacobian matrix. We remark that because these results use a variational inequality formulation, they are still valid if side constraints are included, a fact that can be verified similarly to the argument used in this chapter.

Chapter 4

An Efficient Route Guidance System

The design of route guidance systems faces a well-known dilemma. The approach that theoretically yields the system-optimal traffic pattern may discriminate against some users in favor of others. Proposed alternative models, however, do not directly address the system perspective and may result in inferior performance. In Section 4.3, we propose a novel model and, in Section 4.4, the corresponding algorithms to resolve this dilemma. The essence of this model is that system-optimal routing of traffic flow with explicit integration of user constraints leads to a better performance than the user equilibrium, while simultaneously guaranteeing superior fairness compared to the pure system optimum. The algorithm is based on a method called Partan, which is a revised version of the Frank-Wolfe algorithm. Section 4.5 presents computational results on real-world instances and compare the new approach with the well-established traffic assignment model. Many of the real-world instances that we use were kindly provided by DaimlerChrysler AG, Berlin. Additional instances were retrieved from an online library called *Transportation Network Test Problems* (Bar-Gera 2002). Later, Chapter 5 complements this one by analyzing the route guidance system from a theoretical point of view.

This chapter is based on a research article by Jahn, Möhring, Schulz, and Stier-Moses (2004).

4.1 Route Guidance Systems

Route guidance and information systems, collectively called *Intelligent Transportation Systems*, are designed to assist drivers in making route decisions. Such devices can provide information (e.g., conditions drivers are likely to experience) or give recommendations (e.g., "leave the highway at the next exit and turn right"). We will concentrate on in-vehicle route guidance devices that provide recommendations to drivers. Drivers enter their destinations at the beginning of the trip, and the system computes routes based on digital maps, up-to-date traffic data and current vehicle positions determined with the help of the *Global Positioning System*[1] (Henry, Charbonnier, and Farges 1991). These devices normally use visual and acoustic indicators to aid drivers in following the proposed route. Currently, many cars are already equipped with simple versions of these devices, and with prices going down many more are likely to have one in the not-so-distant future. For that reason, it is widely hoped that route guidance systems can help to alleviate the congestion caused by the still increasing amount of road traffic. Even small improvements can have a significant impact.

Several kinds of in-car navigation systems have been proposed. The simplest devices perform *static* guidance (Bottom 2000); i.e., they work with information that is infrequently updated. The vast majority of the in-car guidance consoles deployed today are of this type. Their main goal is to provide information to drivers who do not know the area well. From an algorithmic point of view, they are straightforward: they only compute shortest paths (or approximations thereof) to the destinations with respect to travel time, geographic distance, or other appropriate measures. Computational challenges for these approaches arise "solely" from the huge size of the underlying road networks (Yang et al. 1991; Chou, Romeijn, and Smith 1998).

More sophisticated route guidance systems make use of information on current conditions in the traffic network. To implement this, one-way—or even better, two-way—communication with a traffic control center must be available. With one-way commu-

[1] The Global Positioning System (GPS) is used to determine the location of the vehicle. A receiver listens to signals generated by satellites, computes angles, and finally performs a triangulation to find its latitude and longitude.

nication, current road conditions are determined through sensors placed in the network and then broadcasted to users, who can use the information to compute realistic shortest paths to their destinations. With bidirectional communication equipment, the traffic control center would receive users' current positions and destinations, allowing it to perform some kind of traffic assignment (not necessarily a user equilibrium). Finally, routes in the assignment are randomly assigned to real drivers and transmitted back to the route guidance devices.

The knowledge of the current conditions is the basis of *reactive* guidance systems (Papageorgiou 1990; Friesz et al. 1993; Ben-Akiva, de Palma, and Kaysi 1996). In other words, the recommendation provided to drivers at any given time is based on a snapshot of the traffic at that time. One of the advantages of reactive guidance is that it can respond quickly to demand changes or incidents because no predictions are used.

The most advanced approach, called *anticipatory* guidance, predicts future demands and traffic conditions and gives recommendations accordingly (e.g., Kaufman, Smith, and Wunderlich 1991; Chen and Underwood 1991; Kaysi, Ben-Akiva, and Koutsopoulos 1993; Ben-Akiva, Cascetta, and Gunn 1995). The issue is how future conditions should be predicted. When market penetration is low, guidance systems can basically ignore their own effect. On the other extreme, when most users are guided and they comply with the guidance, reality is likely to be as predicted. Between the two extremes, the situation is more delicate. These route guidance systems must predict how users will behave (e.g., follow the recommendation or not) to guide traffic in a way that is consistent with the predictions (see Bottom 2000 and the references therein). Otherwise, guidance can fail to achieve the desired objective because recommendations were given making assumptions concerning the future that may not materialize.

According to Bottom (2000), there is no consensus in the community on which of the latter two approaches—reactive or anticipatory—should be used in practice. For the present thesis, we adopt reactive guidance because it is conceptually simpler.

Regardless of the source of network data, route guidance devices still have to compute concrete routes to propose to users. Several systems compute shortest paths, the k shortest paths for some properly chosen parameter k, or Pareto-optimal paths (when multiple criteria are considered simultaneously). Some systems perform these computations online while others perform them in a preprocessing step. In addition, most current route guidance systems implement both user and system optimality, although the bias has always been towards user-optimal traffic patterns (e.g., Mahmassani et al. 1994; Ben-Akiva et al. 1997; Dynasmart 2002). Although system optimality is included in such systems for computing good upper bounds on traffic efficiency, it is not accepted as a realistic option for actual guidance (Mahmassani and Peeta 1993). Indeed, it is well-known that under system-optimal traffic patterns some users may end up traveling longer to allow the system to achieve global efficiency. Of course, it is not likely that many users accept recommendations that are too inefficient with respect to their personal optimal choices. We measure the detriment for users as the ratio of the latency of the recommended path to that of the shortest possible path the user could have taken. This concept, called *unfairness*, will play a central role in this chapter and those to come.

Merchant and Nemhauser (1978) recognized that the assumptions of the traffic assignment problem are unrealistic and proposed to consider a *dynamic* model. Since then, there has been significant effort towards the dynamic analysis of traffic networks (see, e.g., Ben-Akiva 1985; Friesz 1985; Mahmassani and Peeta 1995; Peeta and Ziliaskopoulos 2001 and the references therein). Unlike static traffic assignment, where models and solution methods are well established, the dynamic traffic assignment problem has been studied from several different perspectives with no single generally accepted model or methodology. We refer the reader to the articles by Mahmassani and Peeta (1995) and Peeta and Ziliaskopoulos (2001), which provide a discussion of the inherent difficulties and corresponding solution attempts.

As an example, let us mention that DynaMIT (2002), a simulation-based real-time system to provide travel information, computes k shortest paths beforehand with respect to several static latency functions. Among other measures, it considers free flow travel times, peak-period travel times, geographic lengths, and the number of signalized intersections. Then, performing traffic simulation, it computes the dynamic user equilibrium in which users are restricted to taking only those paths.

For a more comprehensive discussion concerning route guidance and its history, we refer the interested reader to the Ph.D. theses by Kaysi (1992), Peeta (1994), and Bottom (2000).

4.2 A Different Approach

None of the existing or proposed guidance systems takes directly into account the efficiency of the solution they propose (with the exception of system-optimal solutions, which are not implementable because of their unfairness). Thus, the need for integrated algorithms that actually pay attention to the system-wide performance has been recognized (Henry, Charbonnier, and Farges 1991; Beccaria and Bolelli 1991; Kaysi, Ben-Akiva, and de Palma 1995).[2]

As mentioned earlier, the most popular approach is to route drivers according to a user equilibrium. In that way, drivers are routed along their respective lowest-latency paths so there are no paths they would prefer to the ones they are given. While a user equilibrium should satisfy the drivers, as we have seen in Chapters 2 and 3, it does not necessarily minimize the total latency in the system. Another unfavorable property of the user equilibrium is its non-monotonicity with respect to the network's capacity. This is illustrated by the Braess paradox, described in Section 2.4.2.

From a global perspective, e.g., the traffic authority's point of view, it is certainly desirable to explicitly minimize the total travel time by computing a system optimum. In particular, the existing road network could then carry more traffic (Lafortune et al. 1991; Ferris and Ruszczyński 1997). Yet, users' needs have to be taken into account: directly implemented, this policy could route some drivers on unacceptably long paths in order to use shorter paths for many other drivers. In fact, the length of a route in the system optimum can be higher than in user equilibrium, even in the simplistic case of a single OD pair (Roughgarden 2002). This is critical because routes can only be recommended to drivers. It is reasonable to assume that only very few of them would be willing to

[2] Not all agree with this idea. For instance, Hall (1996) says that "The suggestion is that ATIS [Advanced Traveler Information Systems] should not be viewed as a strategy for achieving system optimal traffic distributions. ATIS should instead be viewed first as a service to the public, to improve their confidence and comfort in using the system, and second as a means for steering traffic away from dis-equilibrium behavior and toward user optima that utilize alternate routes where feasible."

sacrifice their own short routes for the benefit of the "community." On the other hand, user acceptance of a route guidance system is important if it is supposed to help in reducing traffic congestion. Therefore, Beccaria and Bolelli (1991) have suggested to:

> find the route guidance strategy which minimizes some global and community criteria with individual needs as constraints.

We adopt a system optimum approach but honor the individual needs by imposing additional constraints to ensure that drivers are assigned to "acceptable" paths only. Note that these constraints are totally different from those analyzed in Chapter 3, this chapter's constraints are used to determine if a path that a user could take would be a valid option or not. More precisely, we introduce the concept of the *normal length* of a path, which can be either its traversal time in the uncongested network, its traversal time in user equilibrium,[3] its geographic distance, or any other appropriate measure. The only condition imposed on the normal length of a path is that it may not depend on the actual flow on the path. Equipped with this definition, we look for a *constrained system optimum* in which no path carrying positive flow between a certain OD pair is allowed to exceed the normal length of a shortest path between the same OD pair by more than a tolerable factor. By doing so, we achieve our primary goal of finding solutions that are fair and efficient at the same time.

The novelty of our approach consists in defining a constrained system optimum with the "right" set of allowable paths. We demonstrate that this model leads to a significantly better utilization of a traffic network than the standard traffic assignment (user equilibrium) and still guarantees fairness similar to that in the user equilibrium. To the best of our knowledge, no other work introduces a constrained system optimum approach that guarantees fairness comparable to that of the ordinary traffic assignment. While we study the method from a computational perspective in this chapter, Chapter 5 analyzes this idea theoretically and provides estimates of the efficiency gain when using constrained system optima instead of user equilibria. In addition, the next chapter presents theoretical results on the fairness of constrained system optima.

[3] Throughout this chapter, we consider the user equilibrium without side constraints, as it was introduced in Chapter 2.

4.3 The Route Guidance Model

We consider a model of reactive route guidance that allows us to work with static flows. While not considering dynamic flows may preclude the direct application to real-world situations, our approach can provide traffic planners with bounds on the total travel time that are more accurate (compared to the ordinary system optimum). Moreover, Sheffi (1985) points out that there are times when traffic exhibits steady-state behavior; e.g., during rush hours. If nothing else, this research is a first step in explicitly incorporating system-wide effects into route guidance systems.

We assume that all drivers use the route guidance system and that they actually follow the recommended routes. Admittedly, this assumption is relatively strong, but this should be considered a first step. Future research will explore the design of *consistent route guidance systems* that optimize efficiency without comprising user acceptance. One way to model a non-perfect market penetration is by considering two classes of users. Some users have access to route guidance devices and follow the recommendations, while the remaining users act selfishly. In this extension, a central question is that of creating a traffic pattern for the guided users that is fair and minimizes the total travel time (for all users, including those without guidance). Along this direction, Roughgarden (2004c) studied how to compute an optimal strategy in a network consisting of a set of parallel links.

In addition to the features of the standard model described in Chapter 2, we consider that arcs in the network have an extra attribute. Each arc $a \in A$ has a normal length $\tau_a \geqslant 0$ that serves as an a priori estimate for its traversal time in the solution we seek. Normal lengths can be chosen to be any metric for the arcs that is fixed in advance. However, their proper choice will allow us to produce solutions with desirable features; we refer the reader to Section 4.5 for details. For a given path $P \in \mathcal{P}$, its normal length is $\tau_P \overset{\text{def}}{=} \sum_{a \in P} \tau_a$.

τ_a

τ_P

We assess the quality of a particular traffic assignment using two criteria. The total travel time in the system matters to the traffic authority while its (un)fairness is of direct importance to users. We have already introduced the former in Chapter 2; let us now discuss the latter.

4.3.1 Measures of Unfairness

Without any centralized control, we would expect flow to be similar to a user equilibrium. Such a flow is well-known to be “fair” in the sense that users between the same OD pair encounter the same delay. However, as we have seen in Chapter 2, a user equilibrium does in general not minimize the total travel time in the system. Our goal is to select more efficient traffic patterns without loosing the fairness property. To make this more precise, let us introduce several notions of *unfairness* of a solution. For a given flow, we define the unfairness of a particular traveler as follows:

Loaded unfairness ratio of her experienced travel time to the experienced travel time of the fastest traveler for the same OD pair, where “experienced travel time” means travel time measured in terms of the current congestion level.

Normal unfairness ratio of the length of her path to the length of the shortest path for the same OD pair, both measured with respect to normal arc lengths.

User equilibrium (UE) unfairness ratio of her experienced travel time to the travel time for the same OD pair in a user equilibrium (which is the same for all users of that OD pair).

Free flow unfairness ratio of her experienced travel time to the length of the fastest path for the same OD pair with respect to free flow travel times.

The respective notion of unfairness for a particular flow is the maximum over all OD pairs of the maximum unfairness of a traveler between that OD pair. More formally, for a given flow x and an equilibrium flow f,

unfairness

$$\begin{aligned}
\text{Loaded unfairness}(x) &\overset{\text{def}}{=} \max\{\ell_Q(x)/\ell_R(x) : Q, R \in \mathcal{P}_k,\ x_Q, x_R > 0,\ k \in K\},\\
\text{Normal unfairness}(x) &\overset{\text{def}}{=} \max\{\tau_Q/\tau_R : Q, R \in \mathcal{P}_k,\ x_Q > 0,\ k \in K\},\\
\text{UE unfairness}(x) &\overset{\text{def}}{=} \max\{\ell_Q(x)/\ell_R(f) : Q, R \in \mathcal{P}_k,\ x_Q > 0,\ f_R > 0,\ k \in K\},
\end{aligned}$$

$$\text{Free flow unfairness}(x) \overset{\text{def}}{=} \max\{\ell_Q(x)/\ell_R(0) : Q, R \in \mathcal{P}_k,\ x_Q > 0,\ k \in K\}.$$

The notions of loaded and normal unfairness are similar. Both compare, using different metrics, the travel times of users to the shortest travel times they could have had. The UE unfairness, introduced by Roughgarden (2002) in the single-commodity context, indicates how the travel times of the solution relate to those in user equilibrium. However, drivers typically do not know the travel times in equilibrium; it is arguably more important to them how their travel times compare to the actual travel times of others. The free flow unfairness measures the degradation of performance that users experience due to the prevalence of congestion effects. Note that the normal unfairness and the loaded unfairness are always greater than or equal to 1, while the UE unfairness and the free flow unfairness can be any nonnegative number.

4.3.2 Problem Formulation

As it is difficult to directly control the loaded unfairness, we will instead impose an upper bound on the normal unfairness and show that by doing so the other notions of unfairness will be small as well. In particular, we consider solutions for which the normal length of any used path between OD pair k is not much greater than that of a shortest s_k-t_k-path (with respect to normal lengths), for all $k \in K$. More specifically, we fix a tolerance factor $\varphi \geqslant 1$ and restrict the normal unfairness to be smaller than φ. In other words, a path $P \in \mathcal{P}_k$ is *feasible* if $\tau_P \leqslant \varphi T_k$. Here, $T_k \overset{\text{def}}{=} \min_{P \in \mathcal{P}_k} \tau_P$ is the normal length of a shortest path between s_k and t_k. If we let $\mathcal{P}_k^\varphi$ denote the set of all feasible paths for OD pair k, we can define the entire set of feasible paths as $\mathcal{P}^\varphi \overset{\text{def}}{=} \bigcup_{k \in K} \mathcal{P}_k^\varphi$.

φ
T_k
$\mathcal{P}_k^\varphi$
$\mathcal{P}^\varphi$

Because route guidance systems eventually have to propose paths to the drivers, our formulation is path-based: there is a decision variable x_P for each path $P \in \mathcal{P}^\varphi$. In fact, it is virtually impossible to model the restriction to feasible paths with the help of a formulation based on arc variables only. Moreover, even if one were (somehow) given an arc flow that has a decomposition into feasible paths, it is NP-hard to compute such a decomposition (Corollary 6.4). In contrast, user equilibria and ordinary system optima can be computed using arc-based formulations; any flow decomposition results in path flows with the desired property.

CSO The constrained system optimum that we propose to use in route guidance systems is an optimal solution to the following min-cost multicommodity flow problem with separable objective function and path constraints:

$$
\text{Problem CSO:} \qquad
\begin{aligned}
&\min && C(x) && && \text{(4.1a)}\\
&\text{s.t.} && \sum_{P\in\mathcal{P}_k^{\varphi}} x_P = d_k && k\in K, && \text{(4.1b)}\\
& && x_P \geqslant 0 && P\in\mathcal{P}^{\varphi}\,. && \text{(4.1c)}
\end{aligned}
$$

In order to guarantee that this problem has a unique solution, we will assume throughout this and the following chapter that $\ell(x)x$ is convex (i.e., ℓ is s-convex) for all $\ell \in \mathcal{L}$. Note that the flow variables are not required to be integral since they describe abstract flow rates. If paths were not restricted to be feasible (i.e., in $\mathcal{P}^{\varphi}$), an optimal solution to this formulation would coincide with an ordinary system optimum.
CSO$^{\varphi}$ We denote by CSO$^{\varphi}$ an optimal solution to the problem with tolerance factor φ.

Figure 4-1 demonstrates the effect of path constraints on the system optimum. One commodity is routed through the road network between two clearly marked terminals. In the picture on the left, we display the (unconstrained) system optimum. The flow is distributed widely over the network in order to avoid high arc flows, which would incur high arc travel times. In the picture on the right, the same demand is routed, but this time with the restriction that the normal length of any used path is at most 10% longer than that of the shortest path (i.e., $\varphi = 1.1$). In this example, the normal length has been chosen to be the geographic distance. Line thickness reflects arc capacity (light gray) and arc usage (black), respectively.

Before we discuss the computational complexity of Problem CSO and algorithms to find a constrained system optimum, let us emphasize that this model is different from traffic assignment formulations with side constraints that we introduced in Chapter 3. As pointed out before, the most commonly considered type of side constraints are explicit bounds on arc flows. Nonetheless, such constraints cannot be used to render

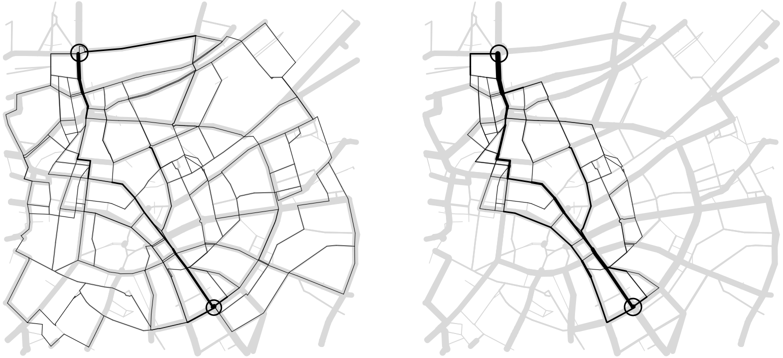


Figure 4-1: System optimum without and with restrictions on the normal length of paths, resp.

certain paths infeasible, as we have argued earlier. Still, path-based multicommodity flow models similar to ours with explicit constraints on the set of allowable paths are frequently used in other application areas. A recent example is the work by Holmberg and Yuan (2003), who study routing problems in telecommunication networks and solve the resulting models by column generation. However, nobody has tried to capture aspects of system optimality and user fairness in a network with congestion effects, as we do.

4.4 Algorithms and Complexity

To solve Problem CSO, we use a variant of the convex combination algorithm of Frank and Wolfe (1956). As it is well-known that the standard Frank-Wolfe algorithm sometimes shows poor convergence (see, e.g., Sheffi 1985; Patriksson 1994; Florian and Hearn 1995), we consider an improved version called *Partan* that was proposed by LeBlanc, Helgason, and Boyce (1985) and further studied by Florian, Guélat, and Spiess (1987) and Arezki and Van Vliet (1990), among others. As we cannot explicitly work with all variables x_P associated with paths $P \in \mathcal{P}^{\varphi}$, because there may be exponentially many, we only generate them when needed. For that reason, our algorithm can be considered to be a column generation method. The application of column generation to the computation of system optima and user equilibria was first studied by

Gibert (1968) and Leventhal, Nemhauser, and Trotter (1973).

For the sake of completeness, let us briefly describe the Frank-Wolfe method.[4] In every iteration, and starting from a current solution, the algorithm solves a linearized version of Problem CSO to determine a feasible descent direction. As the linearization permits the decomposition of the problem by commodities, it is enough to call a subroutine for finding a shortest path in $\mathcal{P}_k^\varphi$ for each commodity $k \in K$. In the subsequent line search, the original nonlinear problem is solved restricted to the line defined by the feasible direction of descent. The algorithm terminates when a certain precision is achieved. To determine when this is the case, the convexity of the objective function is used to derive a lower bound on the value of an optimal solution. It is well known that this algorithm always converges to a global minimum (for convex programs). Partan is based on the same idea, but it performs a more intelligent line search. It determines the descent direction using the results of two consecutive iterations, thereby diminishing the zigzagging effect.

The sub-step of computing a shortest path in $\mathcal{P}_k^\varphi$ is precisely the so-called *constrained shortest path problem*; see Section 4.4.1 below. The only difference between the algorithm we just described and the version of Frank-Wolfe (or Partan) employed for computing user equilibria or system optima is the use of constrained shortest paths instead of regular shortest paths in the solution of the linear subproblems.

Note that other methods like *partial linearization algorithms* or *simplicial decomposition* can also be adapted to our problem. Since we want to make the point that constrained system optima are useful, it was not necessary to implement potentially more efficient algorithms as we can solve relatively large instances within acceptable time limits by using Partan. As others concluded before, for our purpose "…the [Frank-Wolfe] algorithm is considered sufficiently good for practical use" (Patriksson 1994, p. 100). Nevertheless, if one wants to deploy these ideas in a real-time setting, more careful and efficient implementations are needed. We refer the reader to the books by Sheffi (1985), Nagurney (1993), and Patriksson (1994) as well as the chapter by Florian

[4] Jahn, Möhring, Schulz, and Stier-Moses (2002) provided an in-depth description of the implemented algorithms.

and Hearn (1995) for comprehensive overviews of these and many other algorithms.

4.4.1 The Constrained Shortest Path Problem

Let us sketch how the computation of constrained shortest paths—the pricing component of our column generation approach—is carried out. In this subproblem, every arc $a \in A$ has two parameters, a traversal time ℓ_a and a length τ_a. Given an origin-destination pair (s, t), the objective is to compute a quickest path from s to t whose length does not exceed a given bound T. That is, one wants to solve the following problem:

$$\min\{\ell_P : P \text{ is a path from } s \text{ to } t \text{ such that } \tau_P \leqslant T\},$$

where $\ell_P \overset{\text{def}}{=} \sum_{a \in P} \ell_a$ and $\tau_P \overset{\text{def}}{=} \sum_{a \in P} \tau_a$. This problem is NP-hard (Garey and Johnson 1979).

For solving this problem, Aneja and Nair (1978) proposed to use Lagrangian relaxation; Ribeiro and Minoux (1986) added a branch-and-bound scheme. Aneja, Aggarwal, and Nair (1983) extended Dijkstra's algorithm to the case of two objective functions, and Climaco and Martins (1982) used path ranking.

Because of its superior computational efficiency, we implemented the label correcting algorithm of Aneja et al. (1983).[5] The algorithm fans out from the start node s and labels each reached node $v \in N$ with labels of the form $(d_\ell(v), d_\tau(v))$. For each path from s to v that has been detected so far, $d_\ell(v)$ represents its traversal time and $d_\tau(v)$ its distance. During the course of the algorithm, several labels may have to be stored for each node v, namely the Pareto-optimal labels of all paths that have reached it. This labeling algorithm can be interpreted as a special kind of branch-and-bound with a search strategy similar to breadth-first search. Starting from a certain label of v, one obtains lower bounds for the remaining paths from v to t by separately computing ordinary shortest path distances from v to t with respect to travel times ℓ_a and lengths τ_a,

[5] Another promising approach has recently been suggested by Mehlhorn and Ziegelmann (2000). It is based on the Lagrangian relaxation of the dual of an integer linear programming formulation of the constrained shortest path problem.

respectively. If one of these bounds is too large, the label can be dismissed.

4.4.2 Computational Complexity of Computing a Constrained System Optimum

For the sake of completeness, let us also quickly discuss the computational complexity of Problem CSO. Note that it includes as a special case the situation in which all link performance functions are constant; i.e., $\ell_a(x_a) = \ell_a$ for all $a \in A$. Moreover, the set of feasible paths is only given implicitly. Hence, the input dimension is $|A| + |K|$. As computing a constrained shortest path is an NP-hard problem (Garey and Johnson 1979), it is not hard to see that Problem CSO is also NP-hard, even for $|K| = 1$.

4.5 Computational Study

The computational study is divided into three parts. First, we discuss which normal length should be used in practice. Next, we analyze efficiency vs. fairness of solutions for instances that arise from real-world networks. Finally, we briefly report on the performance of the algorithm itself.

The seven instances we used in this study come from two different sources. Four of them represent different parts of the actual road network of the city of Berlin, Germany, and were provided by DaimlerChrysler AG. Their demand rates stem from origin-destination polls conducted in Berlin. The other three come from the *Transportation Network Test Problems* website (Bar-Gera 2002). Table 4.1 shows the specifics of each instance. Instances are listed in increasing order of the product of the number of arcs and the number of commodities. This measure of complexity has been used in the literature (e.g., Holmberg and Yuan 2003), and it indeed corresponds to the ordering with respect to solution times. Instances range from rather small ones, which were included because they are standard in the literature, to fairly large ones.

The algorithm described in Section 4.4 was implemented in C++ using the GCC compiler under Linux; the computing platform was a Pentium IV based computer run-

Table 4.1: Problem instances used in the computational study

Instance Name	Short Name	Source	$\lvert N\rvert$	$\lvert A\rvert$	$\lvert K\rvert$	$\lvert A\rvert \cdot \lvert K\rvert$
Sioux Falls	SF	TNTP	24	76	528	40K
Friedrichshain	F	DC	224	523	506	265K
Winnipeg	W	TNTP	1,067	2,975	4,344	13M
Neukölln	N	DC	1,890	4,040	3,166	13M
Mitte, Prenzlauerberg & Friedrichshain	MPF	DC	975	2,184	9,801	21M
Chicago Sketch	CS	TNTP	933	2,950	83,113	245M
Berlin	B	DC	12,100	19,570	49,689	972M

ning at 2.4 GHz with 1 GB RAM.

4.5.1 Choice of Normal Length

We initially considered three possible ways to define the normal length of an arc: geographic distances, free flow travel times, and travel times when the network is in user equilibrium. Recall that normal lengths can only be static; for instance, it is not possible to consider travel times under the current solution with the methodology described in this dissertation. The advantage of keeping the model simple is a fast algorithm that still produces solutions with small total travel time and low unfairness. It is important to remark that users do not need to know the normal lengths; they are just an artifact of our algorithm to select solutions that are approximately fair.

Geographic distances and free flow travel times are highly correlated; therefore, one cannot expect significant differences between solutions resulting from choosing either one as the normal length. For free flow travel times, our runs establish that the total travel time of user equilibria is smaller than that of constrained system optima when the factor φ is too small, in agreement with the conclusions to be derived theoretically in Chapter 5. Consequently, to obtain an improvement in the total travel time, bigger factors must be considered. However, this gives rise to relatively high unfairness, which is undesired. As an example, consider instance Neukölln. The graph on the left in Figure 4-2 shows the value of the objective function for different tolerance factors φ, for

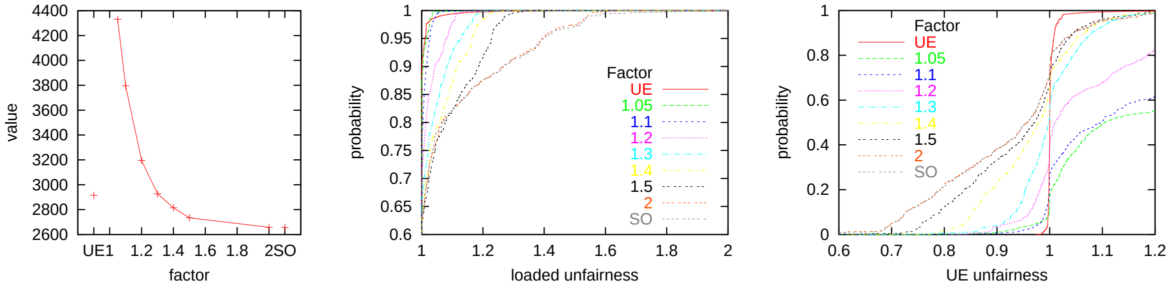


Figure 4-2: Objective values and unfairness distributions for instance Neukölln and normal lengths equal to free flow travel times

the user equilibrium (UE), and for the system optimum (SO). Factors smaller than 1.4 are not helpful because the total travel time of the corresponding solutions is greater than the total travel time in user equilibrium. The other two graphs in Figure 4-2 depict the distribution of unfairness across users for varying tolerance factors; for instance, for factor $\varphi = 1.5$, 80% of all users will experience a loaded unfairness of less than 1.1. This value increases to 1.2 if one considers 90% of all users. For factors greater than 1.5, the distributions are quite similar to that of the system optimum. In the graph on the right, note that for small tolerance factors most users end up traveling longer than they would in user equilibrium. This happens because there are not enough alternative paths between any one OD pair, which explains the poor quality of the solutions under this choice of normal length.

We therefore propose to make use of the travel times in user equilibrium when defining normal arc lengths, which results in high-quality solutions. Indeed, for any factor φ, the user equilibrium itself is a feasible solution to the constrained system optimum problem. Therefore, for all $\varphi \geqslant 1$,

$$C(\mathrm{CSO}^{\varphi}) \leqslant C(\mathrm{UE}),$$

which guarantees that the optimal solution to Problem CSO is never worse than the user equilibrium in terms of the total travel time in the system. The advantage of this normal length definition is that it is flow-dependent; it provides a better indication which paths should be selected. Let us repeat that users do not need to know the user equilibrium; it is just an ingredient for the computation of the constrained system optimum.

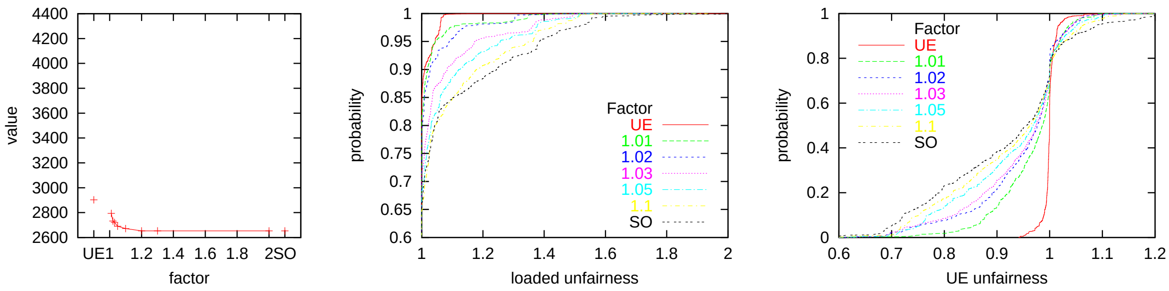


Figure 4-3: Objective values and unfairness distributions for instance Neukölln and normal lengths equal to travel times in user equilibrium

Figure 4-3 displays graphs similar to the ones in Figure 4-2 for this choice of normal length. Most notably, total travel times are distinctively smaller than in equilibrium, while the fraction of users traveling longer than in equilibrium is substantially smaller. We therefore limit our analysis in the sequel to this version of normal length; that is, we assume user equilibrium travel times are used to define normal lengths.

4.5.2 Quality of Constrained System Optima

Tables 4.2 and 4.3 exhibit the output of the algorithm for the instances presented in Table 4.1 and varying tolerance factors. Every row represents one run for the factor reported in the first column. The column *objective value* is the total travel time of the solution; the column *number of paths* contains the number of paths with positive flow, which is an indication of the complexity of the solution. In addition, the tables include the 99th percentiles of the different *unfairness* distributions, the *number of iterations* (one iteration consists of solving the linearized problem and performing the line search; see Section 4.4), and the *time* (in seconds) needed to reach the target optimality gap of 0.5%.

For example, the third row for instance Friedrichshain portrays the attributes of the constrained system optimum with tolerance factor $\varphi = 1.02$. The total travel time is 621, and the users between the 506 different OD pairs are assigned to $1,290$ different paths. The actual travel time for 99% of all users is not more than 65.7% than that of the fastest route between their OD pair. Compared to the user equilibrium, their individual travel times are at most 11.7% higher. Note that the corresponding quantities for the system optimum are 106.3% and 25%, respectively.

Before we interpret the computational results, let us call attention to an apparent anomaly in the rows of Tables 4.2 and 4.3 that correspond to user equilibria. In theory, the normal unfairness, the loaded unfairness, and the UE unfairness should be equal to 1; however, in practice they are obviously not. The reason is that each user equilibrium is computed as the optimal solution of an appropriately defined convex optimization problem as per Beckmann, McGuire, and Winsten (1956). As the algorithm terminates as soon as the value of the current solution is within 0.5% of that of an optimal solution, the solution reported here is merely an approximate user equilibrium. In some sense, the normal unfairness, the loaded unfairness, and the UE unfairness give information about its actual deviation from a user equilibrium. Incidentally, in the derivation of the normal arc lengths, we computed the user equilibrium with higher precision, namely a target optimality gap of 0.01% instead of 0.5%. This explains why the 99th percentiles of normal unfairness, loaded unfairness, and UE unfairness of the user equilibrium are not necessarily equal to one another.

Clearly, the larger the tolerance factor φ the closer is the objective function value of an associated constrained system optimum to that of the unconstrained system optimum, and the higher is its unfairness. On the other hand, smaller tolerance factors lead to "fairer" solutions but also result in larger gaps of the total travel time compared to the unconstrained system optimum. However, we will argue that a carefully chosen tolerance factor strikes a good balance between these two conflicting effects. For the sake of argument, let us consider instance Neukölln with $\varphi = 1.02$.

The gap between the total travel time of $\mathrm{CSO}^{1.02}$ and that of the system optimum is about a third of the gap between the user equilibrium and the system optimum. In fact, the travel time of the system optimum is $2,653$ compared to $2,903$ in user equilibrium and $2,732$ for $\mathrm{CSO}^{1.02}$. Moreover, the travel time of 99% of all users in $\mathrm{CSO}^{1.02}$ is at most 30.4% higher than that of any other traveler (between the same terminals), compared to 55.5% in the system optimum. In other words, the reduction of unfairness amounts roughly to 45%. The numbers are similar for most of the other instances.

Table 4.2: Constrained system optima with different tolerance factors, Part I

factor	total latency	paths	99th unfairness percentile				itera-tions	runtime (sec.)
			normal	loaded	UE	free flow		
Sioux Falls								
UE	7448	989	1.001	1.040	1.031	5.098	31	0
1.01	7263	749	1.001	1.282	1.187	4.908	27	0
1.02	7256	754	1.001	1.258	1.184	4.901	38	0
1.03	7251	758	1.001	1.265	1.195	4.789	34	0
1.05	7239	812	1.035	1.290	1.210	4.749	32	0
1.10	7216	893	1.060	1.283	1.178	4.712	56	0
1.20	7207	984	1.078	1.295	1.168	4.573	46	0
1.30	7201	1129	1.092	1.296	1.170	4.598	64	0
SO	7199	1326	1.092	1.295	1.169	4.599	78	0
Friedrichshain								
UE	682	1713	1.011	1.036	1.062	4.382	27	0
1.01	628	1283	1.008	1.657	1.087	4.163	45	1
1.02	621	1290	1.017	1.652	1.117	4.132	30	1
1.03	613	1515	1.029	1.711	1.094	4.124	42	1
1.05	612	1594	1.046	1.733	1.092	4.130	43	1
1.10	594	1598	1.096	1.929	1.109	3.565	40	1
1.20	591	2080	1.170	2.060	1.177	3.932	74	1
1.30	591	2251	1.213	2.058	1.229	3.948	59	1
SO	591	2631	1.213	2.063	1.250	3.947	63	1
Winnipeg								
UE	857	14633	1.029	1.050	1.047	1.503	16	7
1.01	844	10224	1.009	1.119	1.027	1.429	15	8
1.02	842	11901	1.017	1.123	1.019	1.402	16	8
1.03	842	13123	1.027	1.142	1.027	1.389	18	8
1.05	842	15374	1.043	1.164	1.044	1.409	23	10
1.10	841	17846	1.068	1.192	1.054	1.411	30	12
1.20	841	18619	1.075	1.203	1.058	1.429	33	13
1.30	841	18755	1.078	1.210	1.068	1.458	30	12
SO	841	19331	1.076	1.211	1.066	1.449	33	14
Neukölln								
UE	2903	6744	1.025	1.063	1.053	3.806	21	17
1.01	2794	4380	1.008	1.332	1.084	3.182	15	7
1.02	2732	4700	1.015	1.304	1.072	3.054	17	8
1.03	2721	5665	1.028	1.420	1.070	3.079	18	8
1.05	2690	6427	1.045	1.450	1.099	2.987	22	10
1.10	2672	8755	1.091	1.493	1.125	2.944	47	17
1.20	2653	10018	1.168	1.527	1.179	2.292	54	17
1.30	2653	7983	1.183	1.539	1.193	2.327	48	15
SO	2653	8631	1.187	1.555	1.197	2.335	58	48

Table 4.3: Constrained system optima with different tolerance factors, Part II

factor	total	paths	99th unfairness percentile				itera-	runtime
	latency		normal	loaded	UE	free flow	tions	(sec.)
Mitte, Prenzlauerberg & Friedrichshain								
UE	1845	28091	1.015	1.040	1.032	2.236	16	9
1.01	1771	32476	1.008	1.304	1.051	2.086	25	30
1.02	1762	34618	1.017	1.291	1.045	1.993	25	30
1.03	1755	35392	1.026	1.303	1.045	2.008	24	27
1.05	1733	39320	1.046	1.358	1.060	1.808	26	22
1.10	1727	48968	1.086	1.451	1.083	1.881	29	14
1.20	1726	56687	1.122	1.478	1.122	1.918	37	17
1.30	1726	56304	1.123	1.477	1.124	1.910	35	15
SO	1726	64431	1.127	1.471	1.126	1.921	40	24
Chicago Sketch								
UE	18383	194564	1.017	1.039	1.046	1.592	9	46
1.01	18123	119696	1.007	1.101	1.052	1.543	4	27
1.02	18047	155800	1.016	1.123	1.047	1.509	8	46
1.03	18016	192152	1.025	1.148	1.044	1.492	11	57
1.05	17993	242188	1.043	1.193	1.055	1.499	14	69
1.10	17971	289999	1.072	1.211	1.074	1.504	19	89
1.20	17970	334364	1.081	1.227	1.090	1.496	25	118
1.30	17976	344830	1.085	1.224	1.092	1.498	24	118
SO	17981	331146	1.087	1.238	1.093	1.496	25	117
Berlin								
UE	16223	150922	1.038	1.057	1.058	2.400	15	1584
1.01	16254	98271	1.008	1.135	1.906	3.191	9	904
1.02	15806	142944	1.018	1.214	1.112	2.181	14	1274
1.03	15671	171452	1.028	1.247	1.066	2.058	19	1626
1.05	15632	216328	1.045	1.270	1.060	2.003	29	2247
1.10	15587	257707	1.084	1.333	1.083	2.000	39	2689
1.20	15572	295138	1.126	1.372	1.120	2.016	49	3614
1.30	15565	307050	1.137	1.398	1.128	2.022	52	4184
SO	15544	322687	1.148	1.438	1.135	2.066	56	5512

Figures 4-4 and 4-5 depict the complete unfairness distributions for all instances. Let us again pick Neukölln to highlight typical effects. In $\mathrm{CSO}^{1.02}$, the travel time of just 4.5% of all users is more than 10% than that of the fastest paths of their OD pairs. In contrast, this number is 15.3% for the ordinary system optimum; i.e., one sixth of all drivers experience delays that are significantly above and beyond that of their fellow drivers. Moreover, most users (around 80%) spend less time on the road than they would in equilibrium. Actually, for factor 1.02, only 0.3% of the users travel 10% more than in equilibrium. Compare this number to the 4.6% that travel at least 10% longer under the system optimum.

To facilitate a comparison of the characteristics of constrained system optima with different tolerance factors, Figures 4-6 to 4-12 plot various percentiles of the different notions of unfairness. The two diagrams on top of each figure represent the 95th and 99th percentile, respectively, of the four notions of unfairness. The four remaining graphs correspond to each unfairness definition and show the 95th, 97.5th and 99th percentiles, respectively.

Let us draw attention to some typical effects, and we will once again use instance Neukölln when we need to mention concrete numbers. We first compare the travel times of users in any of the computed route guidance solutions to the length of their shortest paths in the uncongested network (free flow unfairness). It is remarkable that for virtually all tolerance factors in our study, the increase of travel time due to congestion effects is significantly smaller than the corresponding increase in the (approximate) user equilibrium. For example, for Neukölln and the 99th percentile, the free flow unfairness for all constrained system optima is about 3 or lower, while the free flow unfairness of the user equilibrium is 3.8. The significance of this observation is only reinforced by the fact that at equilibrium all users between the same OD pair experience the same delay, while this is not necessarily the case in a constrained system optimum. The second important observation to be made is the strong correlation between the loaded unfairness and the normal unfairness, which is illustrated by the two diagrams in the middle of each figure. Bounding the normal unfairness (a static measure) results in bounded loaded unfairness (a dynamic measure), which explains why our approach is successful.

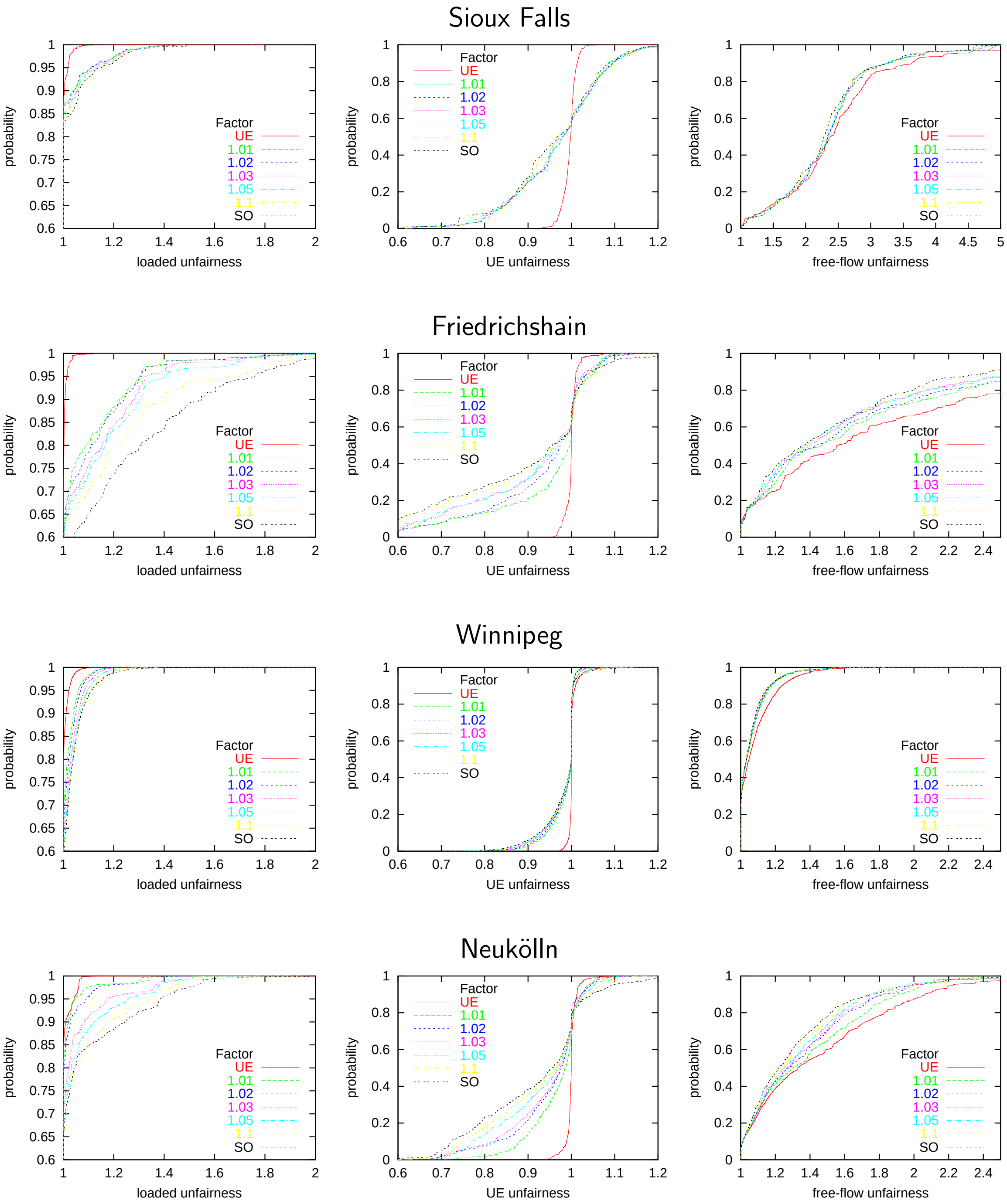
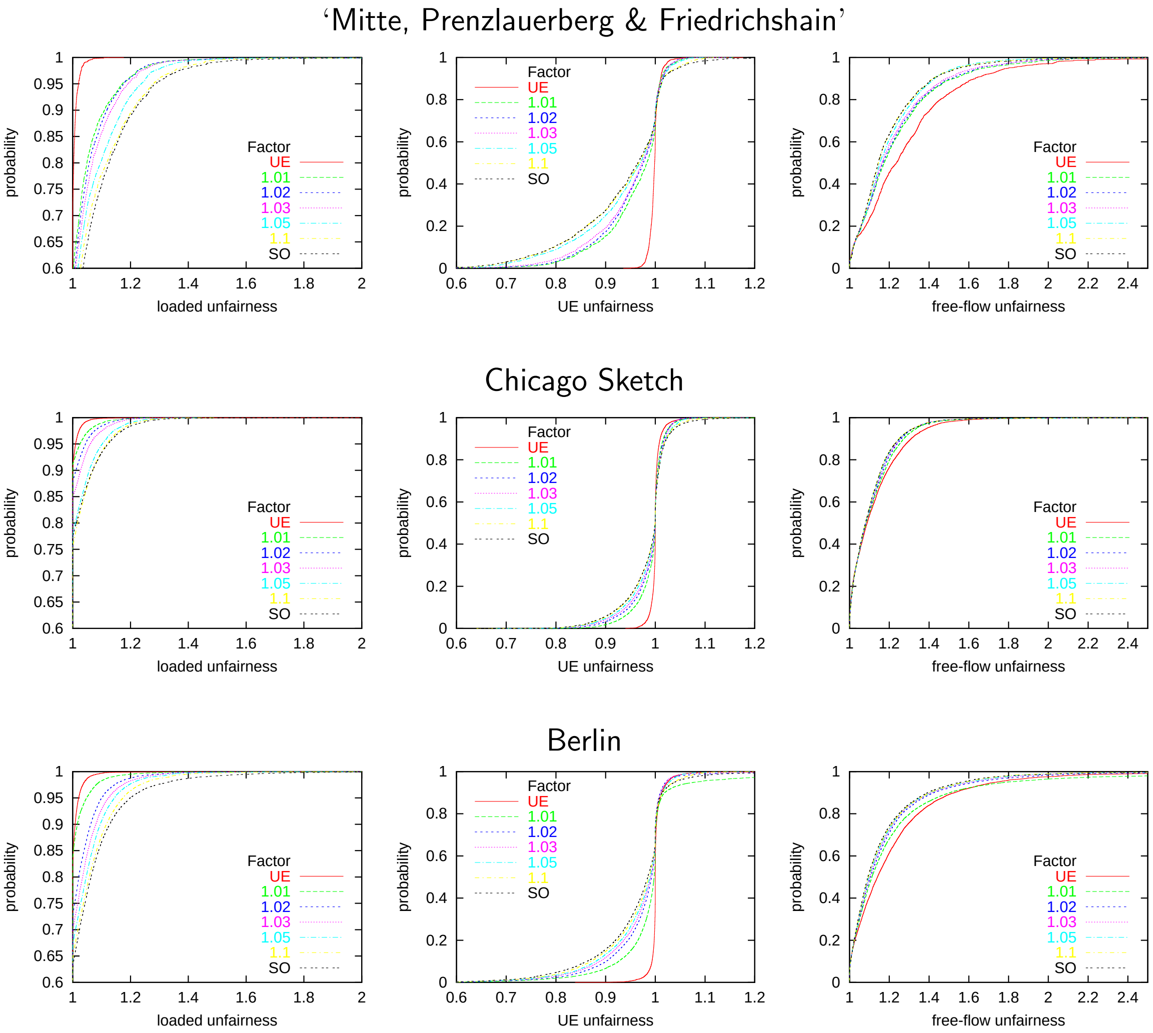


Figure 4-4: Unfairness distributions for various tolerance factors, Part I

 'Mitte, Prenzlauerberg & Friedrichshain'

Chicago Sketch

Berlin

Figure 4-5: Unfairness distributions for various tolerance factors, Part II

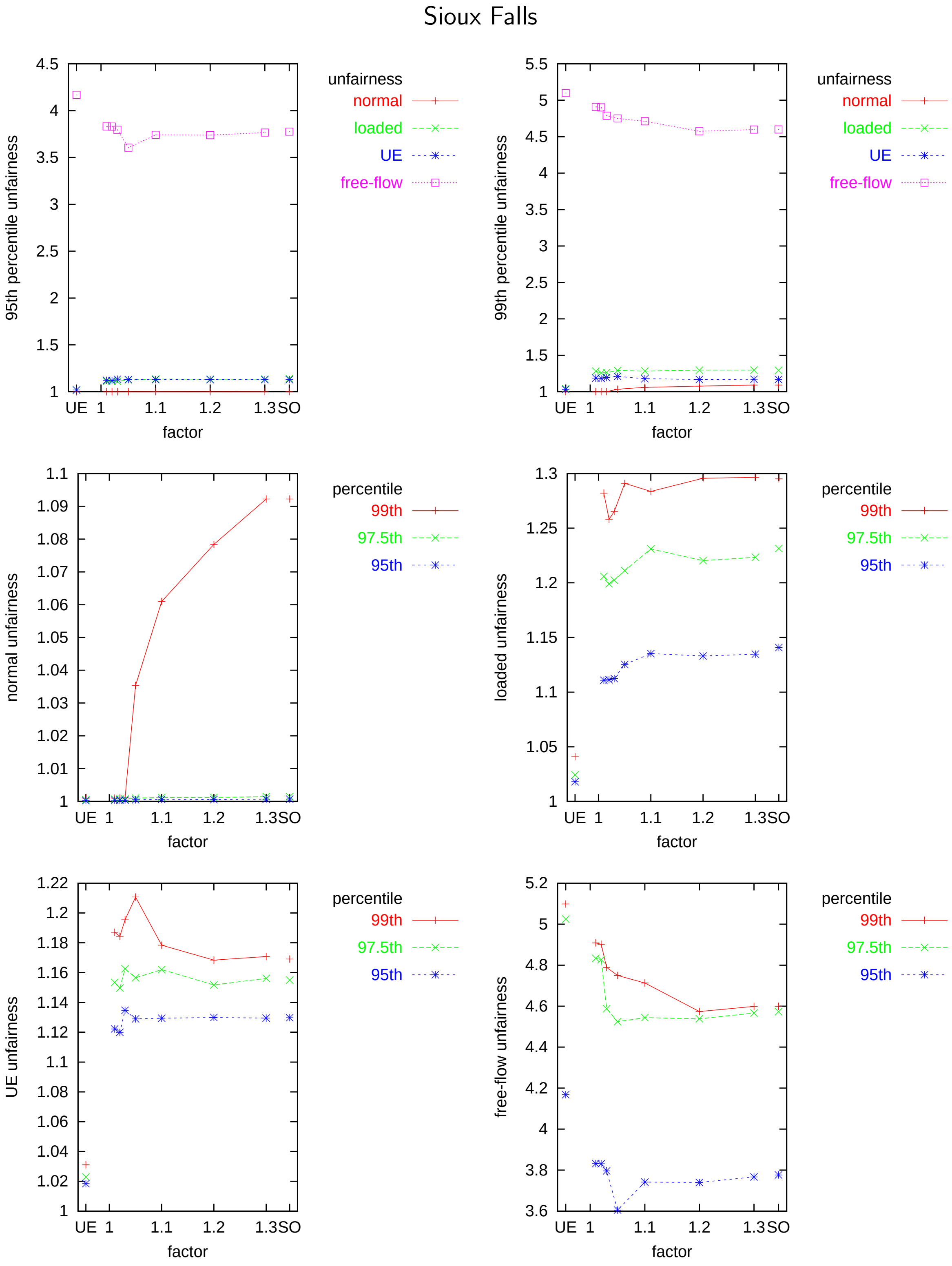


Figure 4-6: Unfairness over the different factors and percentiles for instance Sioux Falls

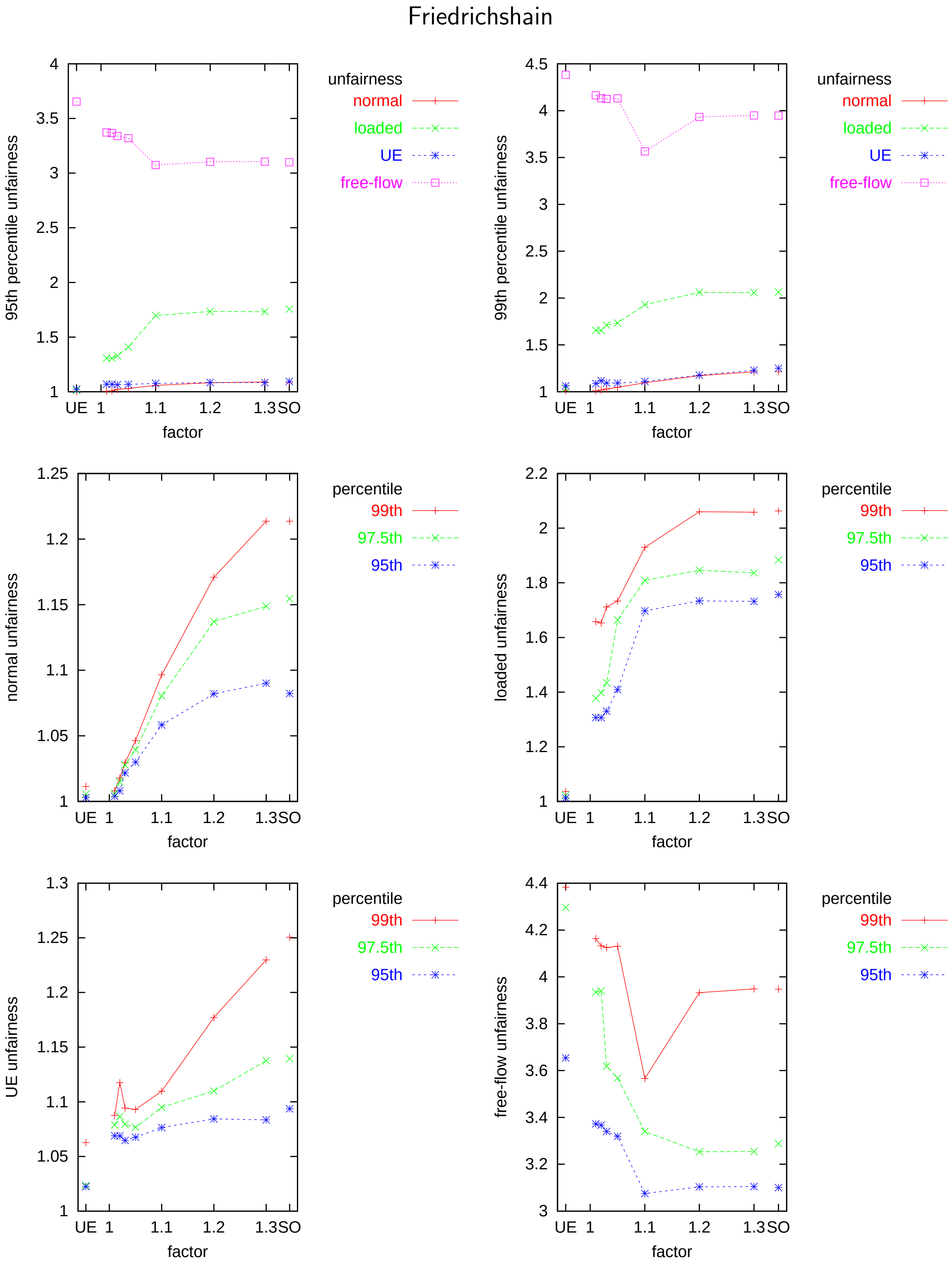


Figure 4-7: Unfairness over the different factors and percentiles for instance Friedrichshain

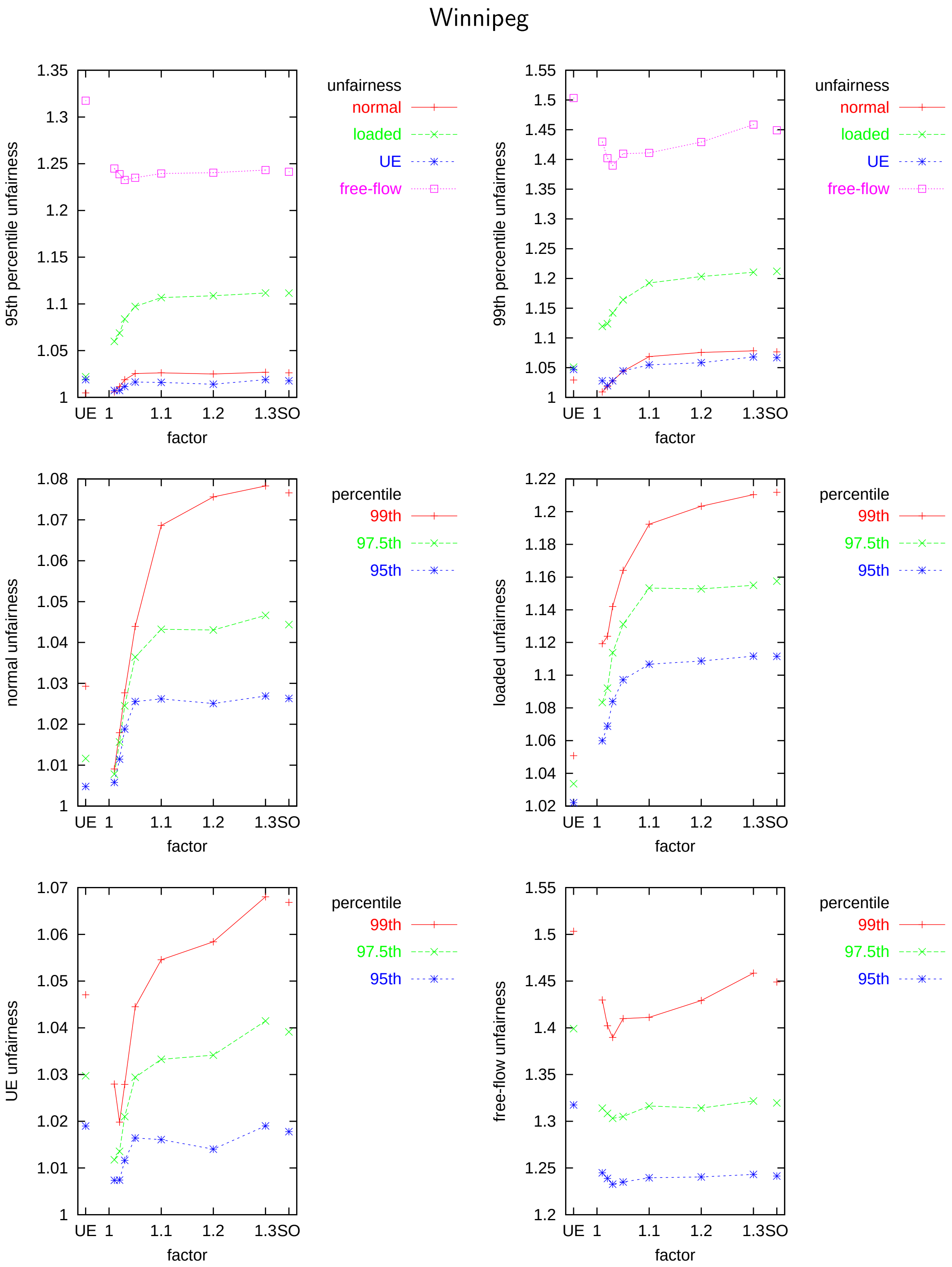


Figure 4-8: Unfairness over the different factors and percentiles for instance Winnipeg

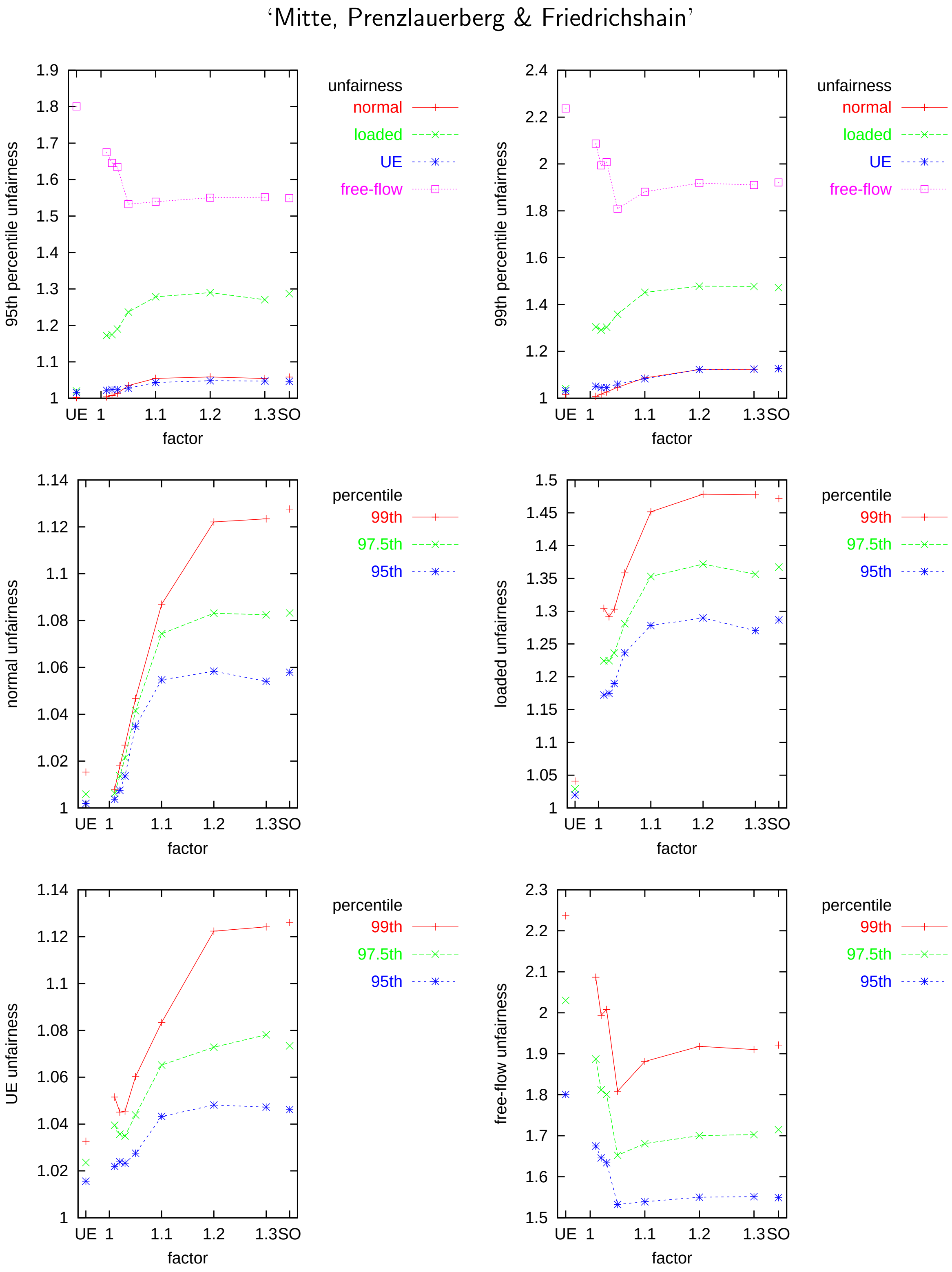


Figure 4-9: Unfairness over the different factors and percentiles for instance 'Mitte, Prenzlauerberg & Friedrichshain'

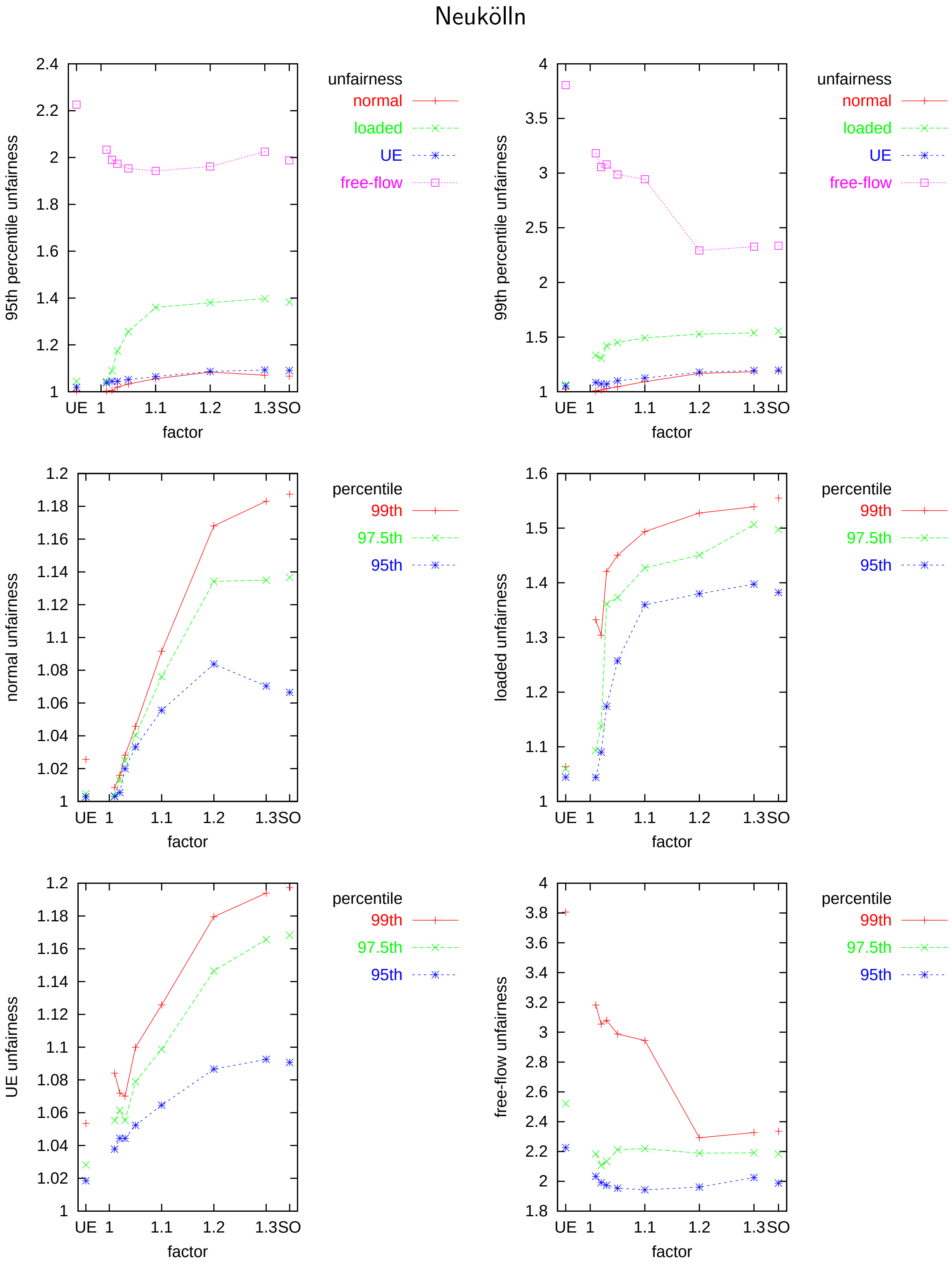


Figure 4-10: Unfairness over the different factors and percentiles for instance Neukölln

Chicago Sketch

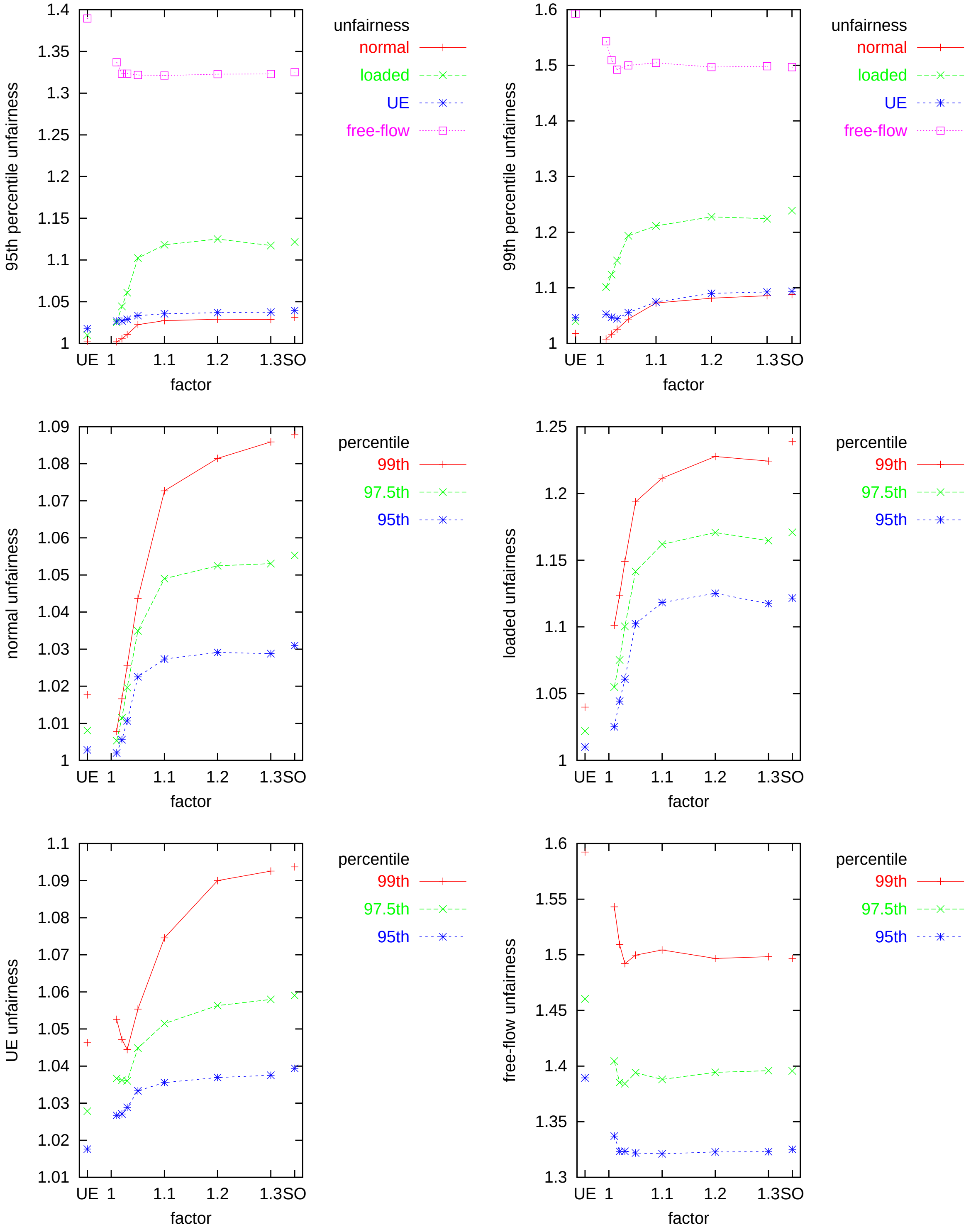


Figure 4-11: Unfairness over the different factors and percentiles for instance Chicago Sketch

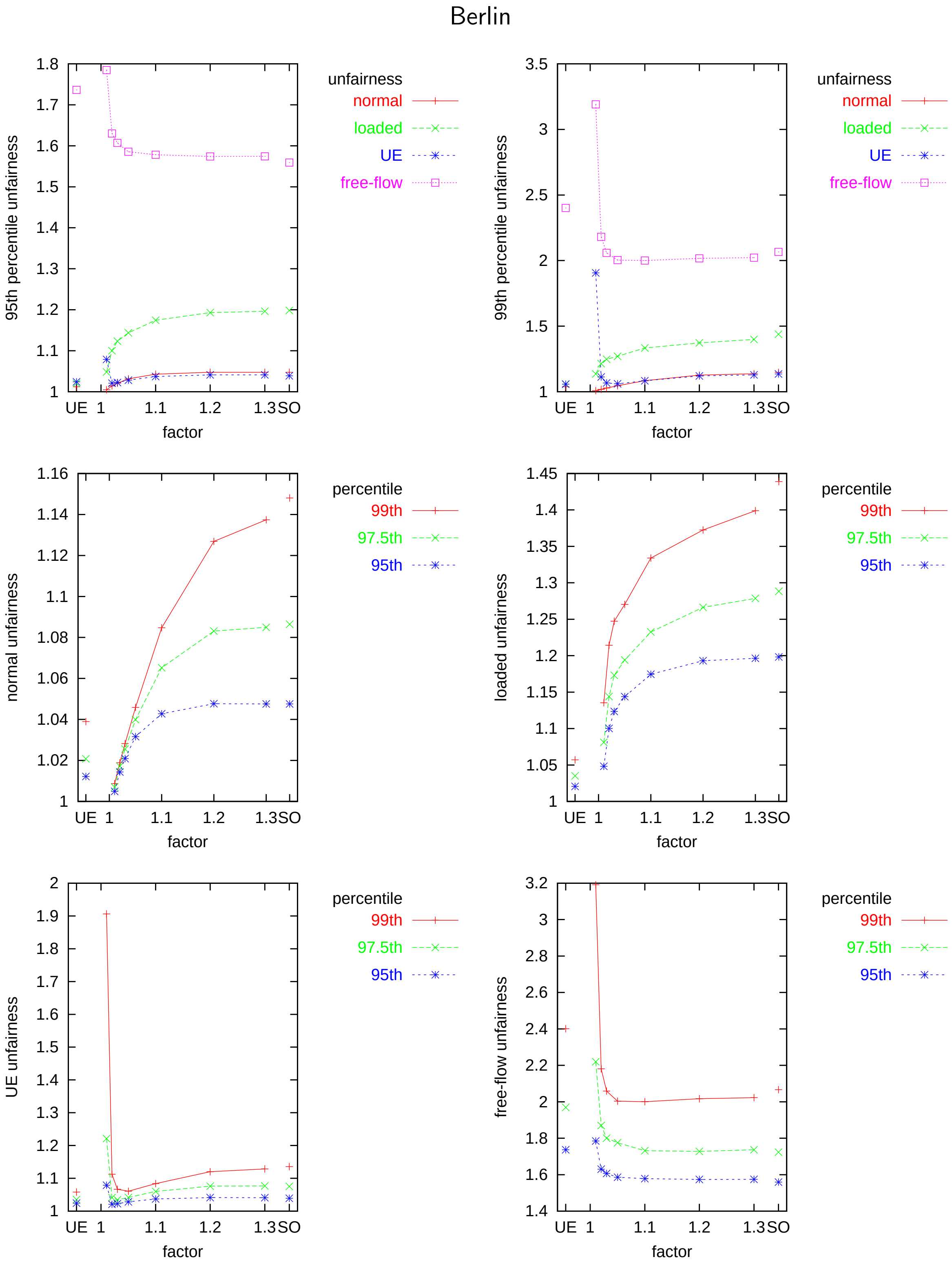


Figure 4-12: Unfairness over the different factors and percentiles for instance Berlin

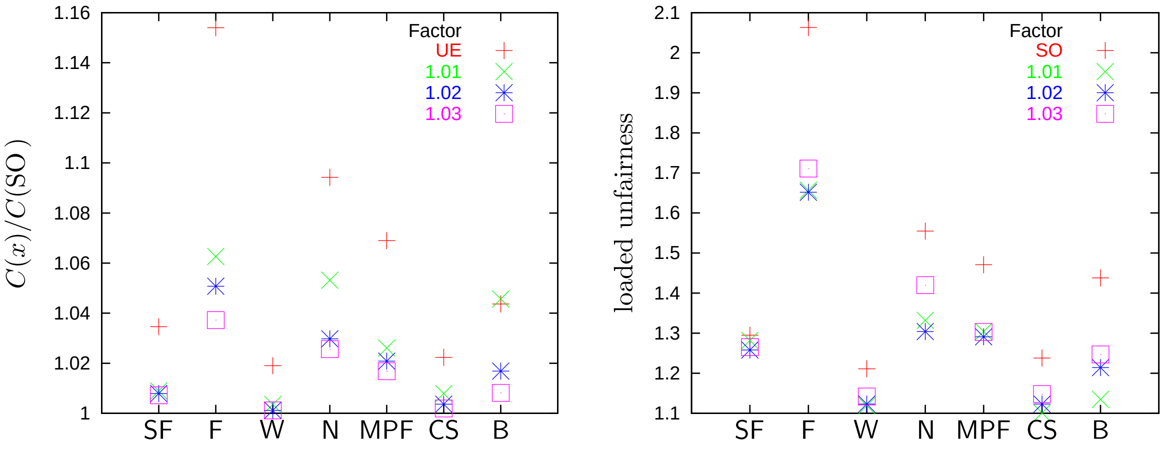


Figure 4-13: Efficiency and loaded unfairness of constrained system optima across all instances. The plot on the left shows the efficiency (the cost of the solution over the cost of the system optimum) of select constrained system optima vs. that of the associated user equilibria; the plot on the right compares the loaded unfairness of the same solutions with that of the corresponding system optima.

Figures 4-13 and 4-14 provide conclusive evidence of the benefits of the solutions we propose; constrained system optima with appropriately chosen tolerance factors bring together the favorable attributes of user equilibria and system optima. In Figure 4-13, we display constrained system optima with tolerance factors close to 1.02 and compare them with the user equilibrium and the unconstrained system optimum, both in terms of efficiency and fairness. Figure 4-14 illustrates the tradeoff between efficiency and fairness achieved by constrained system optima. The graph shows, for each of the instances we studied, system optima (on the left), user equilibria (at the bottom) and the intermediate solutions represented by constrained system optima (in the center). The circled data-points correspond to $\text{CSO}^{1.02}$, for the various instances. In summary, constrained system optima with user equilibrium travel times as normal lengths provide a handle to effectively control the tradeoff between fairness and efficiency.

4.5.3 Performance of the Algorithm

Let us briefly discuss our findings with respect to the running time needed by the algorithm described in Section 4.4. Figure 4-15 shows a detailed study of the effects of

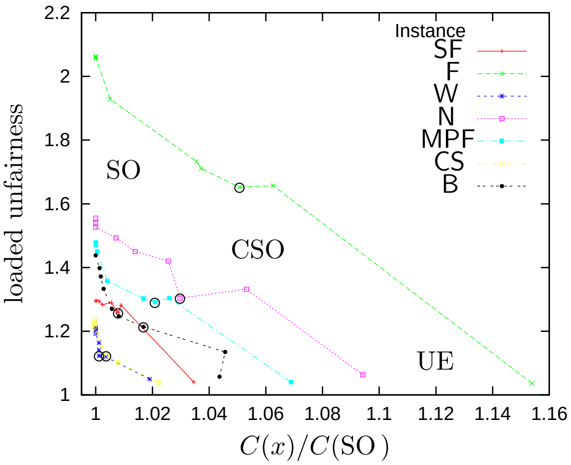


Figure 4-14: Tradeoff between efficiency and unfairness. For all instances, we plot the tradeoff curve between the efficiency (the cost of the solution over the cost of the system optimum) vs. the loaded unfairness. The left area of the graph corresponds to system optima (SO), the lower area corresponds to user equilibria (UE), and the circled data-points (denoted with '$\circ$') correspond to constrained system optima with $\varphi = 1.02$ ($\mathrm{CSO}^{1.02}$).

varying the tolerance factor and the target optimality gap. We only present the results for instances Chicago Sketch and Berlin because they are the largest and hence arguably the most difficult ones. For each selected instance, the figure contains a graph describing the objective function value, another one illustrating the number of iterations, and finally one displaying the computation time (in seconds).

Most notably, the time needed by our algorithm to compute a constrained system optimum is typically not larger than that for computing an unconstrained system optimum, and it is only somewhat larger than that for getting a user equilibrium. In fact, the problem of finding a constrained system optimum becomes computationally more costly with increasing values of the tolerance factor φ. The reason is that the number of allowable paths increases. However, the constrained shortest path subproblems become easier because the normal lengths are less binding. In this trade-off situation, the total work and the number of iterations increase, but the work per iteration decreases. Generally, most of the time is spent on computing constrained shortest paths (which implies that improved algorithms for this subproblem would yield greatly improved overall

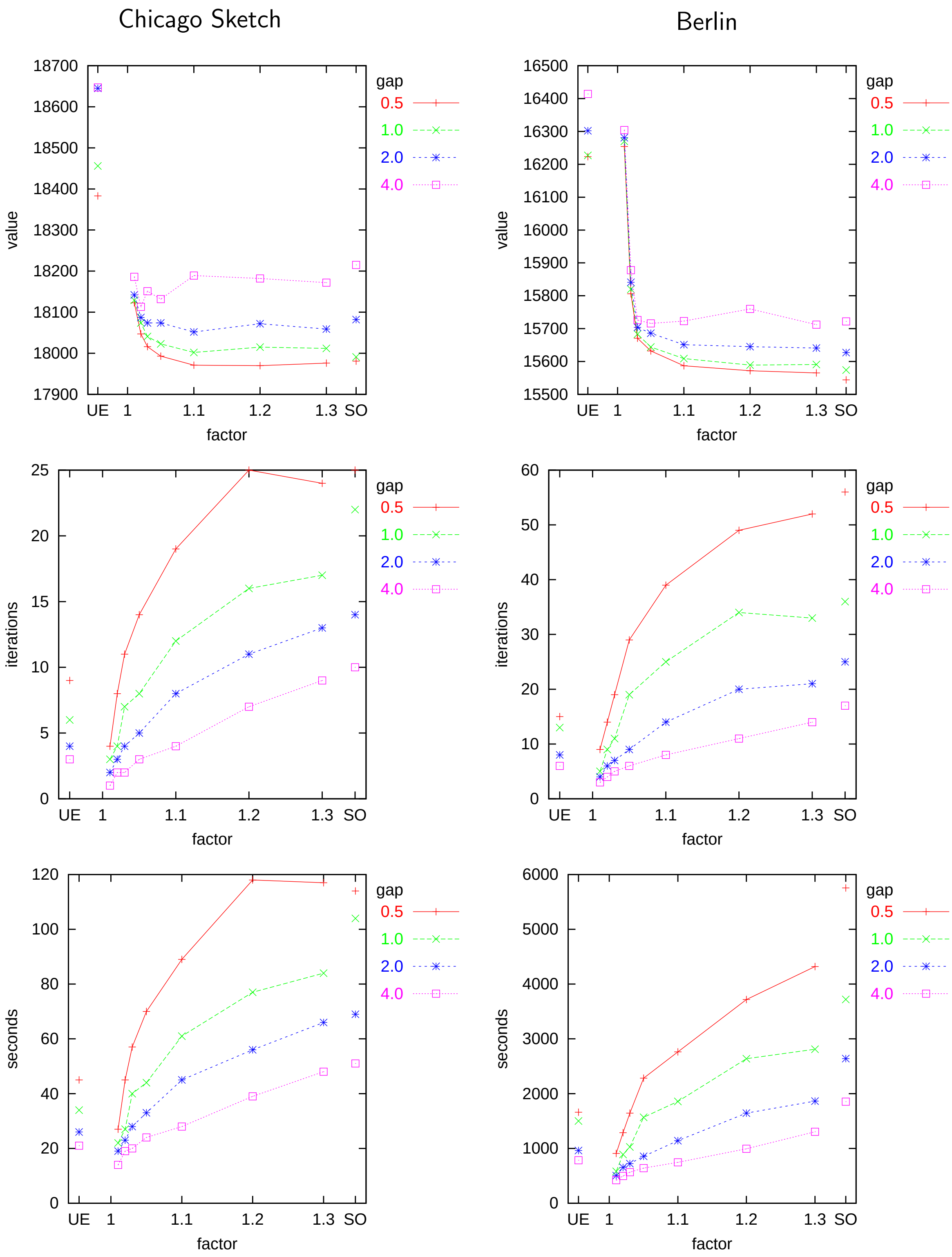


Figure 4-15: Algorithm specifics for various optimality gaps and tolerance factors for instances Chicago Sketch and Berlin

performance).

From our experience, instances with a few thousand nodes, arcs and commodities can be solved on an average PC within minutes. Bigger instances like Berlin take longer but can also be solved without difficulty in less than an hour. Very large instances (e.g., networks with twice as many nodes and arcs as Berlin and with over one million OD pairs) could not be handled mostly due to memory problems resulting from the path-based formulation.

With respect to Partan, we found that the running time is reduced by 30% on average for our target optimality gap of 0.5% when compared to the original version of the Frank-Wolfe method. The reduction is even bigger if just the most difficult instances are considered.

4.6 Discussion

When designing a route guidance system, it is desirable to explicitly aim at reducing the total travel time by putting it into the objective function of the underlying optimization problem. Yet, without further constraints, this would include the possibility that some vehicles are assigned to fairly long paths in order to make the shorter paths available to other drivers. Obviously, this phenomenon would render such a system unacceptable for several drivers, jeopardizing the desired effect of improved system performance.

We have proposed to impose constraints on paths to eliminate lengthy detours. While it may be ideal to explicitly enforce that travel times of recommended routes between the same OD pair do not deviate significantly from each other, our computational results justify the use of a computationally simpler model, in which the deviation is not measured with respect to the actual flow but with respect to a "normal length." Our computational study suggests that the travel time in user equilibrium is an excellent choice for defining the normal length.

In fact, it turns out that this approach offers significant advantages over both the traditionally considered user equilibrium and the system optimum. On the one hand, it guarantees superior fairness for the individual user compared to the system optimum, in which individual travel times between the same OD pair may deviate substantially from each other. On the other hand, the total travel time of a constrained system optimum is still close to that in the ordinary system optimum and thus much better than in user equilibrium. This shows that optimal route guidance with fairness guarantees is in principle feasible. Apart from the proof of concept, we consider our algorithm practical for problems with several thousand nodes, arcs, and commodities. Future work should incorporate the dynamic aspect of traffic and the behavior of unguided users.

Chapter 5

Efficiency and Fairness of the Route Guidance System

We study the route guidance system introduced in Chapter 4 from a theoretical perspective. Recall that normal lengths must be selected to calibrate the system. We show that if the correct normal lengths are used, the constrained system optimum achieves two desired properties: efficiency and fairness. We use the price of anarchy of the system to measure efficiency. In contrast to Chapters 2 and 3, we compare user equilibria to the solutions proposed by the route guidance system. In Section 5.2, we concentrate on the choice of free flow travel times as normal lengths and conclude that a constrained system optimum may be inefficient. Subsequently, Section 5.3 analyzes the case in which normal lengths are defined as user equilibrium travel times. As opposed to the case of free flow travel times, constrained system optima are efficient when these normal lengths are used. Finally, Section 5.4 shows that the loaded unfairness of constrained system optima is bounded from above by a small constant.

This chapter is based on an extended abstract by Schulz and Stier-Moses (2003).

5.1 Introduction

In Chapter 4, we proposed a route guidance system that computes a flow pattern that minimizes the total travel time subject to certain user constraints. These constraints are designed to overcome the inherent problem of system-optimal guidance which may suggest routes that are too long to some users. Indeed, in a system-optimal flow pattern, some users may be routed on considerably longer paths for the benefit of others and hence would typically not follow the route recommendations. (Theorem 5.12 gives a tight upper bound for the unfairness.) User constraints ensure that paths suggested to users are not much longer than shortest paths, both measured with respect to a metric that we referred to as normal length. Recall that normal lengths are intended to model the travel times envisioned for each arc of the network. Although there are no restrictions in how to set them, they have to be defined a priori; in other words, normal lengths cannot depend on the solution that the system will compute. It is important to remark that users need not know the normal lengths; they are merely an artifact to select solutions without detours. Relying on empirical evidence coming from real-world instances, in Chapter 4, we concluded that the flow patterns that the system provides have two desirable properties: the total travel time is close to that of the unconstrained system optimum, and individual users do not experience a considerably larger travel time than the smallest possible.

This chapter complements the previous one by providing a theoretical focus. For this study, we rely on the price of anarchy concept, introduced in Chapter 2 and developed in Chapter 3. Here, though, instead of defining the price of anarchy as the ratio of the total travel time of a user equilibrium to that of an ordinary system optimum, we adopt a more realistic perspective. Flow patterns based on system-optimality only cannot be implemented because of their "unfairness;" therefore, it is more reasonable to measure the price of anarchy with respect to a flow pattern that can potentially be used in practice. For that reason, we measure the efficiency of the route guidance system with the worst-case ratio of the total travel time of an equilibrium to that of a constrained system optimum. In addition, we compare the travel times experienced by different users to each other: a primary goal of a route guidance system is to offer routes with similar travel times to users with the same OD pair. A recommended route is not likely to be accepted if other users with similar origins and destinations obtain routes that are much faster. Even more, as the system assigns users to routes randomly, it is desirable that routes offered on successive days have similar latencies so as to reduce the variance of latencies experienced by individual users. Our results establish that constrained system optima are efficient and fair so long as "correct" normal lengths are selected. This is in agreement with the computational evidence presented earlier.

Recall that there are two aspects that define the quality of a flow. The loaded unfairness of the route assignment is of importance to the users while the total travel time in the system is of importance to the traffic authority (see Section 4.3). Our route guidance system is designed to select the most efficient traffic pattern among those that are not too unfair.

In this chapter, we denote a user equilibrium of a given instance by f, a constrained system optimum with tolerance factor equal to φ by f^{φ}, and an unconstrained system optimum by f^*. The larger the factor φ, the larger the feasible region; consequently, the total travel time $C(f^{\varphi})$ is a nonincreasing function of φ and $C(f^*) \leqslant C(f^{\varphi})$ for all $\varphi \geqslant 1$.

5.2 Free Flow Travel Times as Normal Lengths

In this section, we assume that normal lengths are defined as travel times in the uncongested network; i.e., $\tau_a = \ell_a(0)$ for all $a \in A$. Under this assumption, it turns out that user equilibria have improved performance guarantees when compared to constrained system optima rather than ordinary system optima. This is because constrained system optima with respect to free flow travel times do not perform as well as system optima; moreover, for small values of φ constrained system optima may be even worse than user equilibria. Our theoretical results help to explain the conclusions derived from the computational study of real-world instances developed in Chapter 4.

To analyze the benefits of the proposed concept theoretically, we evaluate the price of anarchy using constrained system optima as a benchmark. Although the original definition of the price of anarchy relies on ordinary system optima, our notion is arguably more realistic because system optima cannot be implemented in practice due to their unfairness. We define the price of anarchy for a tolerance factor φ and a set $\mathcal{L}$ of allowed latency functions as follows:

$\alpha^{\varphi}(\mathcal{L})$

$$\alpha^{\varphi}(\mathcal{L}) \overset{\text{def}}{=} \sup_{\mathcal{I} \in \operatorname{inst}(\mathcal{L})} \quad \frac{C(f_{\mathcal{I}})}{C(f_{\mathcal{I}}^{\varphi})}\,, \tag{5.1}$$

where $\operatorname{inst}(\mathcal{L})$ is the set of instances with latency functions drawn from $\mathcal{L}$, and $f_{\mathcal{I}}$ and $f_{\mathcal{I}}^{\varphi}$ denote the user equilibrium and constrained system optimum of an instance $\mathcal{I}$, respectively. It is immediately clear that $\alpha^{1}(\mathcal{L}) \geqslant 1$ and that $\alpha^{\varphi}(\mathcal{L})$ is nondecreasing as a function of φ. In addition, using Theorem 2.8, we obtain

$$C(f_{\mathcal{I}}) \leqslant \alpha(\mathcal{L})\, C(f_{\mathcal{I}}^{*}) \leqslant \alpha(\mathcal{L})\, C(f_{\mathcal{I}}^{\varphi})\,. \tag{5.2}$$

This implies that $\alpha^{\varphi}(\mathcal{L}) \leqslant \alpha(\mathcal{L})$ for all $\varphi \geqslant 1$. Moreover, for instances with positive minimum normal length T_k for all OD pairs $k \in K$, a constrained system optimum with large tolerance is optimal in the unconstrained sense, i.e., $C(f^{\varphi}) = C(f^{*})$ when φ is sufficiently large.

Let us start our study of the function $\alpha^{\varphi}(\mathcal{L})$ by proving a structural property. We introduce a construction that permits us to modify a given instance for some factor φ so as to obtain an instance for a different factor $\tilde{\varphi}$, but with similar (in)efficiency. Fix a tolerance factor φ and consider an instance $\mathcal{I}$ with large coordination ratio; i.e., for a fixed positive constant ε,

$$\frac{C(f_{\mathcal{I}})}{C(f_{\mathcal{I}}^{\varphi})} \geqslant \alpha^{\varphi}(\mathcal{L}) - \varepsilon\,. \tag{5.3}$$

We construct a new instance $\widetilde{\mathcal{I}}$ equal to $\mathcal{I}$ except for the modifications that follow (see Figure 5-1). The origins in $\widetilde{\mathcal{I}}$ are new vertices $\tilde{s}_k$ for $k \in K$ (instead of s_k), which are connected to s_k with arcs of constant latency M_k, specified below. The natural extension $\tilde{x}$ of a flow x to the new instance is defined as $\tilde{x}_{\tilde{P}} \overset{\text{def}}{=} x_P$ for $P \in \mathcal{P}_k$ and $k \in K$, where $\tilde{P}$ starts from $\tilde{s}_k$ and continues with the original path P. It is not too hard to see that

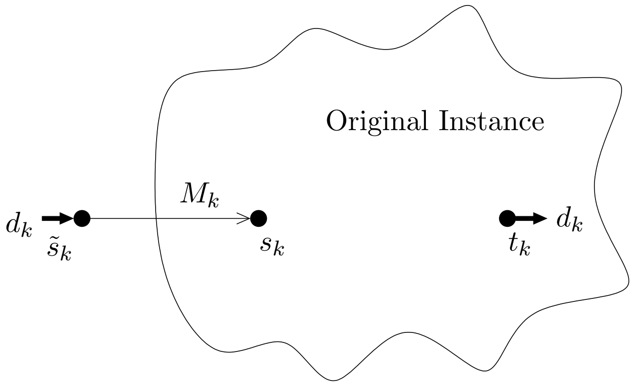


Figure 5-1: Modified instance $\widetilde{\mathcal{I}}$

the extensions $\tilde{f}$ and $\widetilde{f^*}$ of a user equilibrium f and a system optimum f^* of $\mathcal{I}$ are a user equilibrium and a system optimum to instance $\widetilde{\mathcal{I}}$, respectively. The next lemma establishes a relation between the constrained system optima of the two instances.

Lemma 5.1. *Consider a fixed $\tilde{\varphi}$ such that $1 < \tilde{\varphi} < \varphi$, and set $M_k \overset{def}{=} \frac{\varphi - \tilde{\varphi}}{\tilde{\varphi} - 1} T_k$. If f^φ is a φ-constrained system optimum of $\mathcal{I}$, then its natural extension $\widetilde{f^\varphi}$ is a $\tilde{\varphi}$-constrained system optimum of $\widetilde{\mathcal{I}}$.*

Proof. All paths among $\mathcal{P}_k$ that carry flow under f^φ have a normal length between T_k and φT_k. After adding M_k to each of them, their lengths are between $M_k + T_k$ and $M_k + \varphi T_k = \tilde{\varphi}(M_k + T_k)$. It follows that $\widetilde{f^\varphi}$ is a $\tilde{\varphi}$-constrained system optimum. □

Observe that extending a flow x in $\mathcal{I}$ to a flow $\tilde{x}$ in $\widetilde{\mathcal{I}}$ changes its cost by a fixed amount M; that is, $C(\tilde{x}) = M + C(x)$ with $M \overset{\text{def}}{=} \sum_{k \in K} M_k d_k$. Moreover, $M = \frac{\varphi - \tilde{\varphi}}{\tilde{\varphi} - 1} \sum_k T_k d_k \leqslant \frac{\varphi - \tilde{\varphi}}{\tilde{\varphi} - 1} C(x)$, because for this choice of normal lengths, $T_k \leqslant \ell_P(x)$ for all $P \in \mathcal{P}_k$. With this setup, we can now prove that the price of anarchy cannot increase too fast.

Theorem 5.2. *The function $\dfrac{\alpha^\varphi(\mathcal{L})}{\varphi - 1}$ is nonincreasing in φ.*

Proof. Consider the instance $\mathcal{I}$ with large coordination ratio that we selected in (5.3), and let f be a user equilibrium and f^φ be a φ-constrained system optimum. Furthermore, their natural extensions to $\widetilde{\mathcal{I}}$ are referred to as $\tilde{f}$ and $\widetilde{f^\varphi}$, respectively. We bound the price of anarchy of the new instance $\widetilde{\mathcal{I}}$ with that of the original instance $\mathcal{I}$.

$$\alpha^{\tilde{\varphi}}(\mathcal{L}) \geqslant \frac{C(\tilde{f})}{C(\widetilde{f^{\varphi}})} = \frac{M + C(f)}{M + C(f^{\varphi})} \geqslant \frac{C(f)}{\frac{\varphi-1}{\tilde{\varphi}-1} C(f^{\varphi})} \geqslant \tfrac{\tilde{\varphi}-1}{\varphi-1} \left(\alpha^{\varphi}(\mathcal{L}) - \varepsilon\right) \qquad \text{for all } \tilde{\varphi} < \varphi\,,$$

where the first inequality comes from the definition of the price of anarchy, the second uses results from the previous paragraph, and the third is just (5.3). As ε was arbitrary, $\alpha^{\tilde{\varphi}}(\mathcal{L}) \geqslant \frac{\tilde{\varphi}-1}{\varphi-1}\, \alpha^{\varphi}(\mathcal{L})$ for all $\tilde{\varphi} < \varphi$. □

The last theorem implies that the price of anarchy is subadditive as a function of δ, where $\delta \geqslant 0$ is a modified tolerance factor defined as $\varphi - 1$.

Corollary 5.3. *The function $\alpha^{1+\delta}(\mathcal{L})$ is subadditive in δ.*

Proof. Theorem 5.2 implies that for any positive δ_1 and δ_2,

$$\frac{\alpha^{1+\delta_1+\delta_2}(\mathcal{L})}{\delta_1 + \delta_2} \leqslant \frac{\alpha^{1+\delta_1}(\mathcal{L})}{\delta_1} \quad \text{and} \quad \frac{\alpha^{1+\delta_1+\delta_2}(\mathcal{L})}{\delta_1 + \delta_2} \leqslant \frac{\alpha^{1+\delta_2}(\mathcal{L})}{\delta_2}\,.$$

By taking a convex combination, we get that

$$\frac{\alpha^{1+\delta_1+\delta_2}(\mathcal{L})}{\delta_1 + \delta_2} \leqslant \frac{\delta_1}{\delta_1 + \delta_2} \left(\frac{\alpha^{1+\delta_1}(\mathcal{L})}{\delta_1} \right) + \frac{\delta_2}{\delta_1 + \delta_2} \left(\frac{\alpha^{1+\delta_2}(\mathcal{L})}{\delta_2} \right)\,.$$

Therefore, $\alpha^{1+\delta_1+\delta_2}(\mathcal{L}) \leqslant \alpha^{1+\delta_1}(\mathcal{L}) + \alpha^{1+\delta_2}(\mathcal{L})$. □

5.2.1 Bad Instances

In this section, we will characterize instances with high coordination ratios. We call an instance *tight* when its coordination ratio $C(f)/C(f^{\varphi})$ matches the upper bound $\alpha(\mathcal{L})$ shown in (5.2). Here, f and f^{φ} are the user equilibrium and the constrained system optimum of the corresponding instance, respectively. Understanding when an instance is tight might help system administrators design networks with small price of anarchy.

First, we point out conditions that characterize tight instances in the unconstrained case, i.e., $C(f)/C(f^*) = \alpha(\mathcal{L})$. Recall that Pigou's example is tight when latencies are taken from $\mathcal{L}_{\text{lin}}$ (see Sections 2.4.1 and 2.5.1). Roughgarden (2003b) extended Pigou's example to an arbitrary class of latencies $\mathcal{L}$. Indeed, the discussion in Section 3.6 implies that tight instances exist if the supremum in the definition of $\beta(\mathcal{L})$ is attained. By a careful analysis of the proof of Theorem 3.10, we establish conditions that characterize unconstrained instances that are tight. For convenience, we remind the reader that $L_k(f)$ denotes the maximum travel time for users corresponding to OD pair $k \in K$, and that $\ell_a^*(x) = \ell_a(x) + x\ell_a'(x)$ denotes a modified latency function that includes an extra term that accounts for the congestion that users generate; for details check Chapter 2.

Observation 5.4. *Let $\mathcal{L}$ be a family of continuous and nondecreasing latency functions. An unconstrained instance with latency functions drawn from the set $\mathcal{L}$ is tight if and only if the following three conditions are satisfied:*

$$\textit{for all } k \in K \textit{ and } P \in \mathcal{P}_k: \quad f_P^* > 0 \Rightarrow \ell_P(f) = L_k(f)\,, \tag{5.4a}$$

$$\textit{for all } a \in A: \quad f_a^* = \arg\max_{x \geqslant 0} x(\ell_a(f_a) - \ell_a(x))\,, \tag{5.4b}$$

$$\textit{for all } a \in A: \quad \ell_a(f_a)f_a > 0 \Rightarrow \beta(f_a, \ell_a) = \beta(\mathcal{L})\,. \tag{5.4c}$$

Proof. An instance is tight if and only if all inequalities in the proof of Theorem 3.10 are equalities. The conditions correspond to the three inequalities in (3.8), in the order in which they appear. □

Let us make a few remarks related to Observation 5.4:

(i) When latency functions are differentiable and s-convex, we deduce from Condition (5.4b) that $\ell_a^*(f_a^*) = \ell_a(f_a)$. This implies that $\ell_P^*(f^*) = \ell_P(f)$ for all $P \in \mathcal{P}$, and that $L_k^*(f^*) = L_k(f)$ for all $k \in K$. For example, when latencies are linear, the user equilibrium and system optimum of a tight instance must satisfy that $f_a^* = f_a/2$ for all arcs $a \in A$ with the exception of those that have constant latency.

(ii) Condition (5.4a) is redundant because the optimality of f^* and the last remark imply it. Indeed, first-order optimality conditions of Problem SO imply that f^* is optimal if and only if

$$\text{for all } k \in K \text{ and } P \in \mathcal{P}_k: \quad f_P^* > 0 \Rightarrow \ell_P^*(f^*) = L_k^*(f^*)\,,$$

which is equivalent to Condition (5.4a).

(iii) For arcs with strictly increasing latency functions, Condition (5.4b) implies that $f_a^* < f_a$ or $f_a^* = f_a = 0$.

(iv) Assume that the set $\mathcal{L}$ contains a nonconstant function and therefore $\beta(\mathcal{L}) > 0$ (otherwise the price of anarchy is 1 and all instances are tight). If an arc a carries flow in the user equilibrium, it must satisfy $\ell_a(0) = 0$.

Proof. To show the implication, assume that $\ell_a(0) > 0$. Let $\hat{x}$ be the argument that maximizes $x(\ell_a(f_a) - \ell_a(x))$ in the definition of $\beta(f_a, \ell_a)$ given in (3.7). Because $0 < \ell_a(0) \leqslant \ell_a(f_a)$, Condition (5.4c) implies that $\beta(f_a, \ell_a) > 0$ meaning that $\ell_a(\hat{x}) < \ell_a(f_a)$. Therefore,

$$\frac{\hat{x}}{f_a}\left(1 - \frac{\ell_a(\hat{x})}{\ell_a(f_a)}\right) < \frac{\hat{x}}{f_a}\left(1 - \frac{\ell_a(\hat{x}) - \ell_a(0)}{\ell_a(f_a) - \ell_a(0)}\right),$$

and this shows that $\beta(f_a, \ell_a - \ell_a(0)) > \beta(f_a, \ell_a)$, which is a contradiction to Condition (5.4c). □

In Lemma 5.5 below, we shall prove that an instance with latency functions satisfying $\ell_a(0) = 0$ for $a \in A$ cannot be tight. As Remark (iv) prevents all arcs a with $\ell_a(0) > 0$ from carrying flow in a user equilibrium, the lemma implies that a tight instance must have an arc a with $\ell_a(0) > 0$ that carries flow in a system optimum.

Lemma 5.5. *Let $\mathcal{L}$ be a family of continuous and nondecreasing latency functions. Assume further that latencies can only be constant or strictly increasing. If $\ell_a(0) = 0$ for all $a \in A$, the instance cannot be tight.*

Proof. Given the assumption, there are two classes of arcs: those with latencies identically equal to 0 (we refer to them as *0-latency arcs*), and those with strictly increasing latencies. Consider a user equilibrium and a system optimum flow, and one OD pair $k \in K$. Let us define a set $C_k = \{\, i \in N :$ there is a path from s_k to i using 0-latency arcs $\}$. If $t_k \in C_k$, both flows route the demand of OD pair k along a 0-latency path. If this happens for all OD pairs, the instance cannot be tight. Therefore, consider an OD pair $k \in K$ for which $t_k \notin C_k$. Thus, C_k defines a s_k-t_k-cut. Note that all the flow that reaches nodes in C_k has to follow paths that, up to the last node in C_k, consist of 0-latency arcs. By construction of C_k, a forward arc in the cut cannot be a 0-latency arc and a backward arc cannot carry (user or system optimal) flow for OD pair k. The former is obvious; to see the latter, if a backward arc a carried flow, a would be a 0-latency arc because all flow reaching C_k must use paths with latency equal to zero. Thus, its tail would belong to C_k too. We showed that there is no flow of OD pair k entering C_k, and that all the flow exits C_k along non 0-latency arcs. This is a contradiction to Remark (iii) because the sum of the flow on forward arcs of the cut C_k must equal the demand d_k. □

We will use Observation 5.4 and Lemma 5.5 to show that, under mild assumptions, there cannot exist tight instances for the constrained case. Note that this does not prevent $\alpha^{\varphi}(\mathcal{L})$ from being equal to $\alpha(\mathcal{L})$ for some φ.

Theorem 5.6. *Consider an instance with latency functions drawn from $\mathcal{L}$. Assume also that these latencies are either strictly increasing or constant. Then, the coordination ratio $C(f)/C(f^{\varphi}) < \alpha(\mathcal{L})$ for all $\varphi \geqslant 1$, where f denotes the user equilibrium and f^{φ} the φ-constrained system optimum.*

Proof. Suppose that the coordination ratio equals $\alpha(\mathcal{L})$. In that case, f^{φ} is a system optimum because the cost of the system optimum is a lower bound of that of f^{φ} and the coordination ratio cannot be larger than $\alpha(\mathcal{L})$. From Remark (iv), we know that $\ell_a(0) = 0$ for all arcs a with $f_a > 0$. Hence, there is a path joining each OD pair with null free flow travel time. In other words, the normal length T_k has to be 0 for all $k \in K$, which implies that a path is feasible only when its normal length is zero. Therefore, $\ell_a(0) = 0$ for all arcs a with flow in f or f^{φ}, contradicting Lemma 5.5 (we can disregard arcs without flow in both f and f^{φ}). □

We turn our attention to characterizing instances with large coordination ratio for a fixed φ. We say that a path $P \in \mathcal{P}_k$ is *longest* if its normal length τ_P equals the maximum possible φT_k.

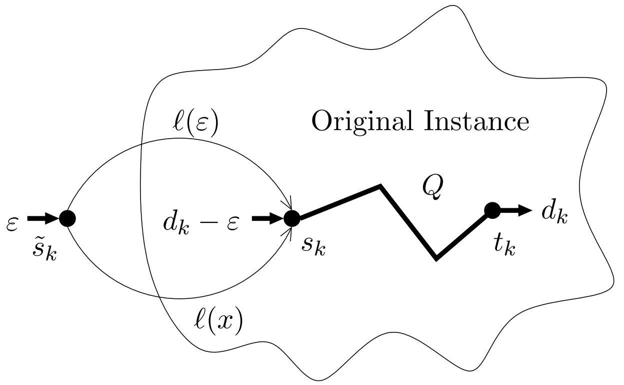


Figure 5-2: Modified instance used in the proof of Theorem 5.7

Theorem 5.7. *Consider a family $\mathcal{L}$ of differentiable and s-convex latency factions. Let $\varphi \geqslant 1$ and f^φ be a φ-constrained system optimum of a given instance with latencies drawn from $\mathcal{L}$. If f^φ routes flow along a path that is not* longest*, the instance can be modified to increase the coordination ratio $C(f)/C(f^\varphi)$.*

Proof. We again use the idea of modifying a network to prove the claim. Suppose that for some $k \in K$ there is path $Q \in \mathcal{P}_k^\varphi$ that is not *longest* such that $f_Q^\varphi > 0$. We will construct an instance with bigger ratio. We modify the given instance by incorporating the tight instance given by Pigou (Section 3.6). Figure 5-2 illustrates the modification. Two parallel arcs connect a new origin $\tilde{s}_k$ to s_k, the origin of path Q. For a small enough number $\varepsilon > 0$, ε units of demand are reassigned from OD pair k to a new OD pair $\tilde{k}$ with terminals $\tilde{s}_k$ and t_k. As before, the latencies of the new arcs are the constant $\ell(\varepsilon)$ and the function $\ell(x)$, where ℓ is a latency function that satisfies $\beta(\varepsilon, \ell) = \beta(\mathcal{L})$.

Consider a path $P \in \mathcal{P}_k^\varphi$. After the modification, there are two possible extensions of P. Denoting the set of paths in the new instance by $\widetilde{\mathcal{P}}$, the path $P_\uparrow \in \widetilde{\mathcal{P}}_k$ (resp. $P_\downarrow$) starts with the constant (resp. nonconstant) arc just added and continues along P. The path $P_\downarrow$ belongs to $\widetilde{\mathcal{P}}_k^\varphi$ because $\tilde{\tau}_{P_\downarrow} = \tau_P$. Although $P_\uparrow$ need not belong to $\widetilde{\mathcal{P}}_k^\varphi$, $Q_\uparrow \in \widetilde{\mathcal{P}}_k^\varphi$ because Q was not *longest.* The user equilibrium and the constrained system optimum of the new instance are simple extensions $\tilde{f}$ and $\tilde{f}^\varphi$ of f and f^φ. We reassign ε units of flow originally in f_Q to $\tilde{f}_{Q_\downarrow}$ and leave the rest as in f. It is straightforward to see that $\tilde{f}$ is feasible and at equilibrium. Similarly, f^φ can be extended to $\tilde{f}^\varphi$ by routing ε units of demand that were originally in Q along the paths $Q_{\downarrow}$ and $Q_{\uparrow}$ in a way that satisfies $\ell^*_{Q_{\downarrow}}(\tilde{f}^{\varphi}) = \ell^*_{Q_{\uparrow}}(\tilde{f}^{\varphi})$. This extension is a constrained system optimum because $\tilde{f}^{\varphi}_a = f^{\varphi}_a$ for all the original arcs a and, therefore, $\ell^*_P(\tilde{f}^{\varphi}) = \ell^*_P(f^{\varphi})$ for all $P \in \widetilde{\mathcal{P}}^{\varphi}$. Computing the total travel times of both flows, the coordination ratio $C(\tilde{f})/C(\tilde{f}^{\varphi})$ equals

$$\frac{C(f) + \ell(\varepsilon)\varepsilon}{C(f^{\varphi}) + \ell(\varepsilon)\varepsilon/\alpha(\mathcal{L})},$$

which is a convex combination of $C(f)/C(f^{\varphi})$ and $\alpha(\mathcal{L})$. As the former is smaller than the latter, the new instance has worse performance. □

5.2.2 Bounds for the Price of Anarchy

In this section, we present upper and lower bounds for the function $\alpha^{\varphi}(\mathcal{L}_{\text{lin}})$. That will allow us to establish that, with free flow normal lengths, constrained system optima are not efficient when compared to user equilibria. We start with an upper bound that improves on $\alpha^{\varphi}(\mathcal{L}_{\text{lin}}) \leqslant \alpha(\mathcal{L}_{\text{lin}}) = 4/3$.

Theorem 5.8. *The price of anarchy $\alpha^{\varphi}(\mathcal{L}_{lin}) \leqslant (2-\varphi)^{-1}$ for all $1 \leqslant \varphi < 2$. In particular, $\alpha^{1}(\mathcal{L}_{lin}) = 1$ and $\alpha^{\varphi}(\mathcal{L}_{lin}) < 4/3$ for $\varphi < 5/4$.*

Proof. Consider a factor $1 \leqslant \varphi < 2$, and let f^{φ} and f be a φ-constrained system optimum and a user equilibrium, respectively. We define the function $h(z) \overset{\text{def}}{=} C(f + z(f^{\varphi} - f))$. Due to the convexity of $C(\cdot)$, $h(1) \geqslant h(0) + h'(0)$. To prove the claim we verify that $h(0) + h'(0) \geqslant (2-\varphi)h(0)$ because then $C(f^{\varphi}) = h(1) \geqslant (2-\varphi)h(0) = (2-\varphi)C(f)$, as required. Now,

$$\begin{aligned} h'(0) &= \sum_a \ell^*_a(f_a)(f^{\varphi}_a - f_a) = \sum_a \left[2\ell_a(f_a) - \ell_a(0)\right](f^{\varphi}_a - f_a) \\ &\geqslant 2\left[\sum_k L_k(f)d_k - \sum_k L_k(f)d_k\right] + \sum_k T_k d_k - \varphi \sum_k T_k d_k \\ &= (1-\varphi)\sum_k T_k d_k \geqslant (1-\varphi)C(f) = (1-\varphi)h(0)\,. \end{aligned}$$

The first inequality comes from the fact that $\ell_P(f) = L_k(f)$ for every $P \in \mathcal{P}_k$ such that

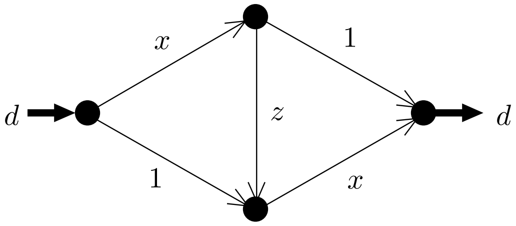


Figure 5-3: Instance used in Lemma 5.9

$f_P > 0$, and $\ell_P(f) \geqslant L_k(f)$ in general. Similarly, $\tau_P \leqslant \varphi T_k$ for every P such that $f_P^\varphi > 0$, and $T_k \leqslant \tau_P$ in general. $\square$

We now give lower bounds for $\alpha^\varphi(\mathcal{L}_{\text{lin}})$ by providing corresponding instances. Although the tight instance discussed in Section 3.6 can be used, a stronger bound can be given with a collection of instances based on the Braess Paradox network (see Section 2.4.2).

Lemma 5.9. *The price of anarchy* $\alpha^\varphi(\mathcal{L}_{lin}) \geqslant 1 + \left(3 + \frac{2}{\varphi - 1}\right)^{-1}$.

Proof. Consider the network depicted in Figure 5-3, based on the Braess Paradox network, where $z \geqslant 0$ is a constant and the demand between the terminals is $d \geqslant 0$. Maximizing the coordination ratio over z and d, we obtain the claim. Indeed, the system optimum and user equilibrium are:

$$C(f^*) = \begin{cases} 2d^2 + zd & \text{if } d \leqslant \frac{1-z}{2} \\ (2-z)d - (1-z)^2/2 & \text{if } \frac{1-z}{2} \leqslant d \leqslant 1-z \\ d^2/2 + d & \text{if } 1-z \leqslant d \,, \end{cases}$$

$$C(f) = \begin{cases} 2d^2 + zd & \text{if } d \leqslant 1-z \\ (2-z)d & \text{if } 1-z \leqslant d \leqslant 2(1-z) \\ d^2/2 + d & \text{if } 2(1-z) \leqslant d \,. \end{cases}$$

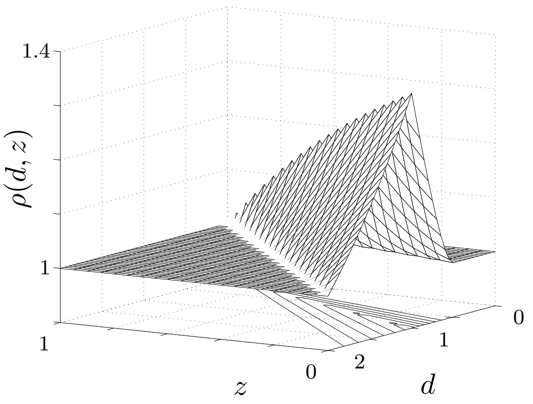


Figure 5-4: The function $\rho(d, z)$

For the constrained system optimum there are two possibilities, either just the shortest path is feasible or all the three paths are. If the later happens, the constrained system optimum is optimal. Therefore,

$$C(f^{\varphi}) = \begin{cases} 2d^2 + zd & \text{if } z < 1 \text{ and } \varphi < \frac{1}{z} \\ \frac{d^2}{2} + d & \text{if } z > 1 \text{ and } \varphi < z \\ C(f^*) & \text{otherwise}\,. \end{cases}$$

We define the coordination ratio for the set of parameters d, z and φ:

$$\rho(d, z, \varphi) \overset{\text{def}}{=} \frac{C(f)}{C(f^{\varphi})}\,.$$

Notice that as a function of φ, $\rho(d, z, \varphi)$ may only take two values depending if φ is small or large. Using the solutions computed above, we can evaluate ρ for any d, z and φ. It turns out that when φ is small, the coordination ratio is bounded from above by 1 for all d and z, thus we will concentrate on the case in which φ is large enough. That case occurs when either $z < 1$ and $\varphi \geqslant \frac{1}{z}$, or when $z > 1$ and $\varphi \geqslant z$. Figure 5-4 plots the values of the coordination ratio as given by the following formula:

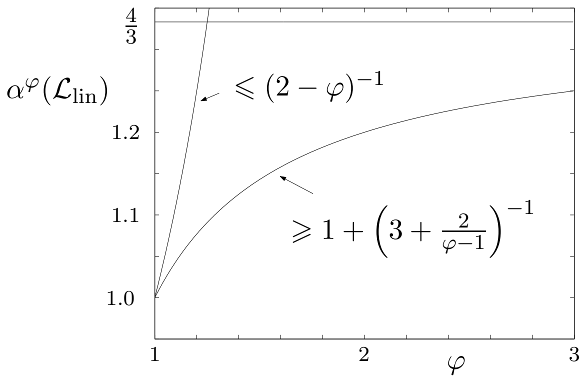


Figure 5-5: Bounds for $\alpha^{\varphi}(\mathcal{L}_{\text{lin}})$

$$\rho(d,z) \overset{\text{def}}{=} \begin{cases} 1 & \text{if } d \leqslant \frac{1-z}{2} \\ \frac{2d^2+zd}{(2-z)d-(1-z)^2/2} & \text{if } \frac{1-\varphi}{2} \leqslant d \leqslant 1-z \\ \frac{(2-z)d}{d^2/2+d} & \text{if } 1-z \leqslant d \leqslant 2(1-z) \\ 1 & \text{if } 2(1-z) \leqslant d\,. \end{cases}$$

The best lower bound can be computed by maximizing the value of ρ. Indeed,

$$\alpha^{\varphi}(\mathcal{L}_{\text{lin}}) \geqslant \max\{\rho(d,z) : \tfrac{1}{\varphi} \leqslant z \leqslant \varphi \text{ and } d \geqslant 0\}\,.$$

We compute the solution of that problem by first solving the maximization for a fixed z. Let $\rho(z) \overset{\text{def}}{=} \max\{\rho(d,z) : d \geqslant 0\}$. For values of z bigger than 1, $\rho(z) = 1$. More interestingly, when $z < 1$,

$$\rho(z) = \rho(1-z,z) = \frac{(2-z)(1-z)}{(1-z)^2/2 + (1-z)} = 2\,\frac{2-z}{3-z}\,.$$

Then, as $\rho(z)$ is nonincreasing, the optimal solution is achieved at $z = 1/\varphi$, from where the claim follows. □

Figure 5-5 summarizes the bounds for $\alpha^{\varphi}(\mathcal{L}_{\text{lin}})$ that we obtained in the last two results. The main conclusion is that $\alpha^\varphi(\mathcal{L}_{\text{lin}})$ is close to 1 when φ is close to 1. Therefore, we need not compute the exact price of anarchy to establish that constrained system optima are almost as bad as user equilibria in the worst case.

5.3 User Equilibrium Travel Times as Normal Lengths

In this section, we assume that normal lengths are set equal to the travel times experienced in a user equilibrium. In Chapter 4, we concluded that these normal lengths are the "correct" choice due to superior empirical performance compared to free flow travel times. Now, we establish that a theoretical analysis supports the same conclusion. The improvement with respect to free flow normal lengths comes from the fact that UE normal lengths depend on the demand.

A user equilibrium f is a feasible solution to Problem CSO because all paths used in f are feasible. Therefore, the constrained system optimum f^φ satisfies

$$C(f^\varphi) \leqslant C(f) \quad \text{for all } \varphi \geqslant 1\,. \tag{5.5}$$

As in the previous section, we obtain a lower bound for the function $\alpha^\varphi(\mathcal{L})$ by providing an appropriate instance. In this case, though, the lower bound matches the upper bound.

Lemma 5.10. *The price of anarchy $\alpha^\varphi(\mathcal{L}) = \alpha(\mathcal{L})$ for all $\varphi \geqslant 1$.*

Proof. Consider the tight instance introduced in Section 3.6 in which a demand of d units must be routed along two parallel arcs with latencies equal to $\ell(d)$ and $\ell(x)$, respectively. At equilibrium both paths have latency $\ell(d)$. Hence, regardless of the value of φ, the system optimum is a φ-constrained system optimum. The claim follows by taking the supremum over $\ell \in \mathcal{L}$. □

The lemma implies that the worst-case coordination ratio is the same whether it is defined with respect to the system optimum or the constrained system optimum. The two bounds presented in (5.2) and (5.5) may be tight. Indeed, we have two extreme examples for $\varphi = 1$. The proof of Lemma 5.10 describes an instance satisfying $C(f^*) = C(f^1) = C(f)/\alpha(\mathcal{L})$. For the second instance, add a small constant $\varepsilon > 0$ to the latency of the first arc. The constrained system optimum coincides with the user equilibrium and, therefore, $C(f) = C(f^1) \approx \alpha(\mathcal{L})C(f^*)$.

Finally, let us remark that Theorem 5.7, proved before for free flow normal lengths, is also valid when normal lengths are set to user equilibrium travel times. To see that, it is enough to note that at equilibrium the lengths of the two new arcs equal $\ell(\varepsilon)$ and therefore $Q_\downarrow$ and $Q_\uparrow$ belong to $\widetilde{\mathcal{P}}^\varphi$.

Observation 5.11. *Let $\varphi \geqslant 1$ and f^φ be a φ-constrained system optimum of a given instance with latencies drawn from $\mathcal{L}$. If f^φ routes flow along a path that is not* longest*, the instance can be modified to increase the coordination ratio $C(f)/C(f^\varphi)$.*

5.4 Fairness

As we mentioned earlier, a typical argument against using the system optimum in the design of route guidance devices for traffic assignment is that, in general, it assigns some drivers to unacceptably long paths in order to use shorter paths for most other drivers. This section presents results related to the unfairness of system optima and constrained system optima. In this section, we work with arbitrary normal lengths, unless explicitly pointed out.

The following theorem quantifies the severity of this effect by characterizing the loaded unfairness of the system optimum. It turns out that there is a relation to earlier work by Roughgarden (2002), who compared the maximum latency of a system optimum in a single-sink single-source network to the latency of a user equilibrium (i.e., the UE unfairness, see Section 4.3.1). He showed that for a given class of latency functions $\mathcal{L}$, this ratio is bounded from above by $\gamma(\mathcal{L})$, which is defined to be the smallest value that satisfies $\ell^*(x) \leqslant \gamma(\mathcal{L})\ell(x)$ for all $\ell \in \mathcal{L}$ and all $x \geqslant 0$. For example, it is easy to see that $\gamma(\{$ polynomials of degree p with positive coefficients $\}) = p + 1$. We prove that the

$\gamma(\mathcal{L})$

loaded unfairness of a system optimum is in fact bounded by the same constant, even for general instances with multiple commodities. The same result was independently obtained by Roughgarden (2003a).

Theorem 5.12. *Let $\mathcal{L}$ be a family of differentiable and nondecreasing latency functions. If f^* denotes a system optimum in a multicommodity flow network with arc latency functions drawn from a class $\mathcal{L}$, the loaded unfairness of f^* is bounded from above by $\gamma(\mathcal{L})$.*

Proof. Using the definitions of ℓ^* and $\gamma(\mathcal{L})$, it is clear that $\ell_a(x) \leqslant \ell^*_a(x) \leqslant \gamma(\mathcal{L})\ell_a(x)$ for all $x \geqslant 0$. The first-order optimality conditions of Problem SO introduced in Section 2.2 imply that $\ell^*_P(f^*) = L^*_k(f^*)$ for all $P \in \mathcal{P}_k$ such that $f^*_P > 0$ (see Proposition 2.5). Therefore, for all paths $P \in \mathcal{P}_k$ carrying flow,

$$\frac{L^*_k(f^*)}{\gamma(\mathcal{L})} \leqslant \ell_P(f^*) \leqslant L^*_k(f^*)\,,$$

from where we deduce that $\ell_Q(f^*)/\ell_R(f^*) \leqslant \gamma(\mathcal{L})$ for all $Q, R \in \mathcal{P}^*_k$ with positive flow. □

An immediate corollary is that users in a system optimum of an s-t-network (i.e., a network with a single source and a single sink) cannot travel too much.

Corollary 5.13. *Let $\mathcal{L}$ be a family of differentiable and nondecreasing latency functions. If f^* denotes a system optimum of a single-source single-sink network with arc latency functions drawn from a class $\mathcal{L}$, then $L(f^*) \leqslant \gamma(\mathcal{L})L(x)$ for any feasible flow x.*

Proof. Obviously, $C(f^*) \leqslant C(x)$ and, therefore, $\min\{\ell_P(f^*) : P \in \mathcal{P}_k, f^*_P > 0\} \leqslant L(x)$. The bound of Theorem 5.12 implies the claim. □

As we mentioned, Roughgarden (2002) proved that, for the single-source single-sink case, $L(f^*) \leqslant \gamma(\mathcal{L})L(f)$. This result also follows by plugging in the user equilibrium in Corollary 5.13. Roughgarden used that bound to argue that system optima are not too unfair because users do not experience a much bigger travel time than they would without control. In addition, it follows that when normal lengths are user equilibrium

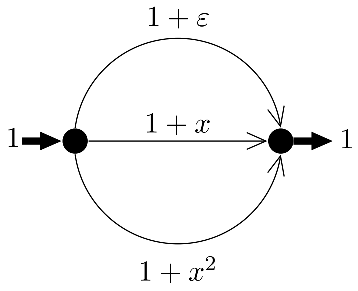


Figure 5-6: The unfairness may decrease when φ increases (free flow travel times case)

travel times, constrained system optima coincide with system optima for $\varphi \geqslant \gamma(\mathcal{L})$, under the assumption that the instance has a single OD pair.

It is straightforward to extend Theorem 5.12 to constrained system optima. Notice that the following result does not assume any particular definition of normal lengths.

Corollary 5.14. *For arbitrary normal lengths, the loaded unfairness of a constrained system optimum is bounded from above by $\gamma(\mathcal{L})$.*

Proof. As first-order optimality conditions for a constrained system optimum are similar to those of an ordinary system optimum, the exact proof of Theorem 5.12 can be repeated by replacing 'system optimum' by 'constrained system optimum,' and $\mathcal{P}_k$ by $\mathcal{P}_k^{\varphi}$. □

In Section 6.4, we present an instance that shows that the bounds given in Theorem 5.12 and Corollary 5.14 are tight. Nevertheless, in practice these bounds are loose and constrained system optima are fairer than the unconstrained counterpart, as the extensive experimentation developed in Chapter 4 shows. Notice that Corollary 5.14 does not imply that the loaded unfairness of constrained system optima with factor φ is nondecreasing as a function of φ. We give two examples, one for each of the definitions of normal lengths.

When normal lengths are free flow travel times, consider the instance depicted in Figure 5-6. That instance has a unit demand, and two terminals connected by three arcs with latencies $1+\varepsilon$, $1+x$ and $1+x^2$, respectively. A constrained system optimum with $\varphi = 1$ can only route flow in the last two arcs and therefore has loaded unfairness strictly larger than 1. Now, for $\varphi \geqslant 1+\varepsilon$, all arcs can be used and it easy to see that

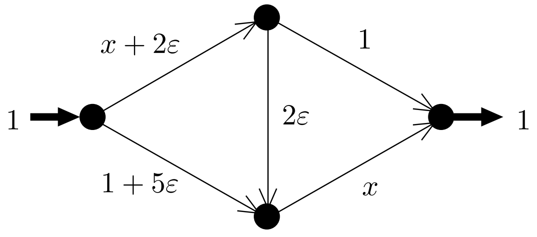


Figure 5-7: The unfairness may decrease when φ increases (user equilibrium travel times case)

the loaded unfairness approaches 1 when $\varepsilon \to 0$.

For the case when normal lengths are user equilibrium travel times, consider the instance shown in Figure 5-7. There are five arcs a, b, c, d and e with latencies $x + 2\varepsilon$, 1, 2ε, $1 + 5\varepsilon$, and x, respectively. The user equilibrium routes flow only along paths ab and ace; at equilibrium the path de is too long to carry flow. Therefore, the constrained system optimum f^1 can only use paths ab and ace, and its unfairness is $\frac{4+4\varepsilon}{3+6\varepsilon}$. Instead, when $\varphi \geqslant \frac{2+3\varepsilon}{2+2\varepsilon}$, the constrained system optimum f^φ can use all three paths. In that case, it routes the flow along ab and cd, and its unfairness is $\frac{6+17\varepsilon}{6+11\varepsilon}$. For small enough values of ε, the unfairness of the constrained system optimum with $\varphi = 1$ is arbitrarily close to 4/3 while the unfairness for a large enough tolerance factor φ is arbitrarily close to 1.

Finally, we show that if a system-optimal solution uses paths that are short with respect to zero flow normal lengths, they must also be short with respect to experienced travel times.

Theorem 5.15. *Consider an instance with linear latencies. Let $P \in \mathcal{P}_k$ be a path such that $f^*_P > 0$ and satisfying that $\ell_P(0) \leqslant \varepsilon\, \overline{C}_k(f^*)$ for some $\varepsilon \geqslant 0$. Then, the experienced travel time $\ell_P(f^*)$ is bounded from above by $(1 + \varepsilon/2)\overline{C}_k(f^*)$.*

Proof. Using Proposition 2.5 and the fact that latencies are linear,

$$\ell_P(f^*) = \ell^*_P(f^*) - \sum_{a \in P} q_a f^*_a = \left(\sum_{Q \in \mathcal{P}_k} \frac{f^*_Q \ell^*_Q(f^*)}{d_k} \right) - (\ell_P(f^*) - \ell_P(0))$$

$$\leqslant \left(\sum_{Q \in \mathcal{P}_k} 2 \frac{f_Q^* \ell_Q(f^*)}{d_k} \right) - \ell_P(f^*) + \varepsilon \overline{C}_k(f^*) \,.$$

Therefore, $2\ell_P(f^*) \leqslant (2+\varepsilon)\overline{C}_k(f^*)$. □

Chapter 6

The Maximum Latency Problem

In this chapter, we study the problem of minimizing the maximum latency of flows in networks with congestion. After giving an example of the problem in Section 6.2, Section 6.3 shows that it is NP-hard, even when all arc latency functions are linear and there is a single source and sink. Section 6.4 proves that an optimal flow and an equilibrium flow share a desirable property: all flow-carrying paths have the same length; i.e., these solutions are "fair," which is in general not true for the optimal flow in networks with nonlinear latency functions. Section 6.5 argues that the price of anarchy is bounded. That is, the maximum latency of a user equilibrium is within a constant factor of that of an optimal solution. In contrast, we present a family of instances that shows that the price of anarchy is unbounded for instances with multiple sources and a single sink, even in networks with linear latencies. Finally, we show that an s-t-flow that is optimal with respect to the average latency objective is near optimal for the maximum latency objective, and it is close to being fair. Conversely, the average latency of a flow minimizing the maximum latency is also within a constant factor of that of a flow minimizing the average latency.

This chapter is based on a research article by Correa, Schulz, and Stier-Moses (2004a).

6.1 Introduction

While in previous chapters we concentrated on the objective of minimizing the total latency of a flow, in this chapter we broaden our view and consider other objectives. Depending on the concrete circumstances, operators of computer networks, traffic control centers, or building designers can pursue a variety of system objectives when planning their systems. For instance, they might elect to minimize the average latency, they might aim at minimizing the maximum latency, or they might try to ensure that users between the same origin-destination pair experience essentially the same latency. In fact, the ideal solution would be simultaneously optimal or near optimal with respect to all three objectives.

For linear latencies, we prove the existence of an s-t-flow that is at the same time optimal for two of the three objectives while its average latency is within a factor of 4/3 of that of an optimum. As attractive as this solution might be, we also show that it is NP-hard to compute. Moreover, there is a surprising difference between linear and nonlinear latency functions. Namely, this particular flow remains optimal with respect to the maximum latency and near optimal with respect to the average latency, but it does in general not guarantee that different users face the same latency. However, an optimal s-t-flow for the average latency objective can be computed in polynomial time, and we show that the latency of any one user is within a constant factor of that of any other user. In particular, the maximum latency is within the same constant factor of the maximum latency of an optimal solution to the latter objective. This constant factor only depends on the class of allowable latency functions. For instance, its value is 2 for the case of linear latencies.

The objective of minimizing the maximum latency is common in the network routing and evacuation literature. The telecommunications network model analyzed by Koutsoupias and Papadimitriou (1999) described in Section 2.5 is similar in nature to that of this chapter. However, their model is not comparable to ours because they work with a finite number of players and consider mixed strategies. In contrast, in our setting every player just controls an infinitesimal amount of flow, making mixed strategies essentially irrelevant; moreover, we work with arbitrary networks.

Starting from the early article by Ford and Fulkerson (1958), most articles on evacuation problems consider constant travel times; we refer the reader to the articles by Chalmet, Francis, and Saunders (1982), Burkard, Dlaska, and Klinz (1993), Hoppe and Tardos (1994, 2000), and Fleischer and Skutella (2003b) for more details. See also the surveys by Aronson (1989) and Powell, Jaillet, and Odoni (1995). One exception is the work of Köhler and Skutella (2002) that considers a dynamic quickest flow problem with load-dependent transit times. They established strong NP-hardness, and they also provided an approximation algorithm by considering the average of a flow over time, which is a static flow. Köhler, Langkau, and Skutella (2002) propose to use time-expanded networks to derive approximation algorithms for a similar problem.

Another important application, if one considers this objective, can be found in supply chains. For instance, a product can only be assembled when all the required components arrive at the production facility. Actually, the maximum latency for this application is nothing else than the lead time to get the parts.

Recall that we assumed that latency functions are nonnegative and nondecreasing. We call a feasible flow that minimizes the maximum latency $L(\cdot)$ a *min-max flow* and denote it by $\hat{f}$. The *maximum latency problem* consists of finding a min-max flow. $\hat{f}$ Although the main goal of this chapter is studying this problem, we still refer to the optimal solution with respect to the average travel time $\overline{C}(\cdot)$ (or $C(\cdot)$) as the *system optimum* and denote it by f^*.

While a user equilibrium can be computed in polynomial time (because it results from the minimization of a convex function over a polytope; see Theorem 2.6), and so can a system optimum if ℓ_a is s-convex for all arcs $a \in A$, we show in Section 6.3 that computing a min-max flow is NP-hard. This result still holds if all latencies are linear and there is a single OD pair.

Recall that Sections 2.5 and 3.5 bound the inefficiency of user equilibria in terms of the average latency. It turns out that equilibria are also bounded with respect to the maximum latency objective. Indeed, latencies encountered by different users of the same origin-destination pair are the same. The user equilibrium therefore represents another

Table 6.1: Summary of results for single-source single-sink networks with linear latency functions

	maximum latency		average latency		unfairness	
min-max flow	1		4/3	Thm. 6.10	1	Thm. 6.5
system optimum	2	Thm. 6.9	1		2	Thm. 5.12
user equilibrium	4/3	Thm. 6.6	4/3	Thm. 2.9	1	

flow that can be computed in polynomial time and that is optimal or close to optimal for the three objectives introduced earlier.

We should point out that among the multiple (nonequivalent) definitions of unfairness introduced in Chapter 4, in the context of Section 6.4 we adopt that of loaded unfairness, the unfairness concept that comes from the competition between different agents. In light of that, sometimes throughout this chapter we may just call it "unfairness." As pointed out earlier, a user equilibrium has unfairness equal to 1 by construction. In Section 6.4, we establish a somewhat surprising result: When latencies are linear and there is a single source and a single sink, there exist min-max flows that are fair.

Finally, in Section 6.5, we show that in the single-source single-sink case under arbitrary latency functions, there actually exist solutions that are simultaneously optimal or near optimal with respect to all three criteria (maximum latency, average latency, and unfairness). In fact, this property is shared by the system optimum, the user equilibrium, and—to some extent—the min-max flow, albeit with different bounds. Let us remark that Jarvis and Ratliff (1982) showed that, in the dynamic case, a flow that minimizes the average latency also minimizes the maximum latency. Although that is not exactly true in our model, there is a big correlation between the objective values of system optima, min-max flows, and user equilibria.

Table 6.1 presents the bounds obtained for the three criteria in the single-source single-sink case with linear latencies. The first entry in each cell represents a worst-case bound on the ratio of the value of the flow associated with the corresponding row to the value of an optimal flow for the objective function denoted by the corresponding column. The second entry refers to the theorem in which the respective result is proved.

All bounds are tight, as examples provided after each theorem demonstrate. The bound of 4/3 on the ratio of the average latency of the user equilibrium to that of the system optimum was first proved by Roughgarden and Tardos (2002); we already gave a simpler proof in Theorem 2.9. Weitz (2001) observed first that this bound carries forward to the maximum latency objective for the case of only one source and sink; we present a generalization of this observation to multicommodity flows in Theorem 6.6. Roughgarden (2004b) gave a tight bound for the single-source single-sink case that depends on the size of the network.

An important consequence of these results is that computing a user equilibrium or a system optimum constitutes a constant-factor approximation algorithm for the NP-hard maximum latency problem. On the other hand, Weitz also presented a family of examples that showed that user equilibria can be arbitrarily bad in multiple commodity networks. We show that already in networks with multiple sources and a single sink, the ratio of the maximum latency of a user equilibrium to that of the min-max flow is not bounded by a constant, even with linear latency functions.

6.2 An Example

Let us start by presenting the solutions to the maximum latency problem for the examples introduced in Section 2.4. In Pigou's instance, every feasible flow has maximum latency equal to 1, making the problem trivial. More interestingly, it is not difficult to see that in Braess' instance, the unique min-max flow coincides with the system optimum. The latter example puts in evidence that a min-max flow may not use shortest paths because the path that goes through the 'middle' arc has travel time equal to 1, as opposed to 3/2, i.e., the travel time of every single user. Moreover, this indicates that a straightforward greedy algorithm that incrementally routes flow along shortest paths does not work. That algorithm would have ended up with the unique user equilibrium.

We use the following instance to compare the three flows we indicated, as well as their objective values. The example, depicted on the top of Figure 6-1, is an instance with one OD pair and quadratic latency functions. The flow has to be shipped through two equal

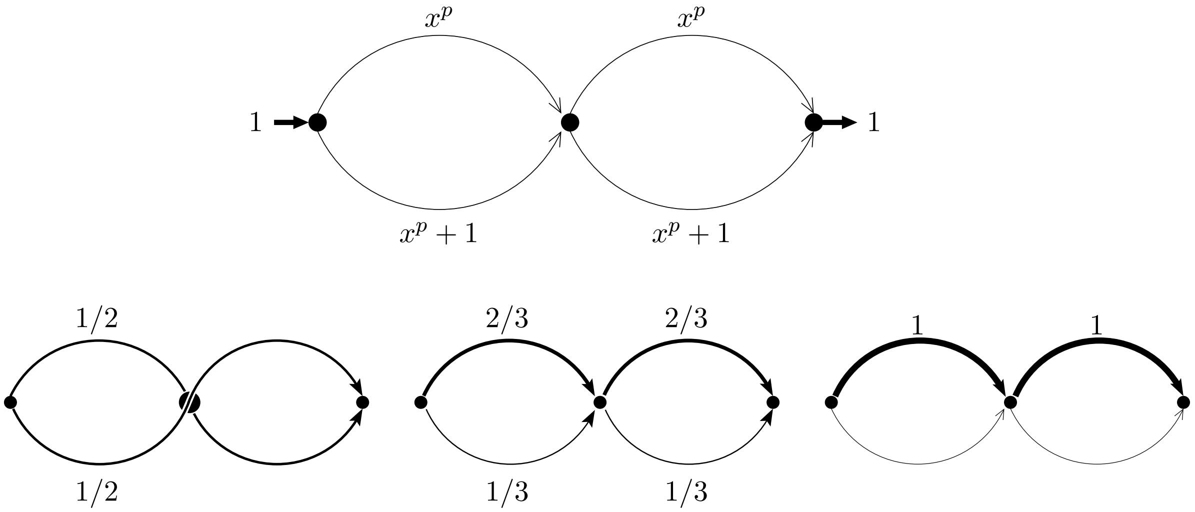


Figure 6-1: An example of the maximum latency problem (*top*) and comparison of a min-max flow, a system optimum, and a user equilibrium (*bottom, from left to right*, respectively)

stages with two parallel arcs each. At the bottom, the figure shows a min-max flow, a system optimum, and user equilibrium, optimal solutions with respect to the maximum travel time, the average travel time, and the unfairness, respectively. Table 6.2 displays the values of the mentioned solutions. Notice that the figure shows the user equilibrium and the system optimum as a flow on arcs. As we mentioned in Chapter 2, any path decomposition provides a correct solution. However, not all of those solutions have the same unfairness and maximum latency; where needed, the table indicates two values coming from the two different decompositions that are possible. The min-max flow, instead, is given as a flow on paths. This is necessary because presenting a flow on arcs is not enough to solve the problem, as we are going to show in Section 6.3. Note that although the unfairness of the min-max flow is 1, we are going to see in Section 6.4 that we can modify this instance to produce an instance that features high unfairness.

This example shows that the objective values of the different solutions are similar. Section 6.5 is devoted to proving guarantees similar to those presented in Table 6.1 for general sets of latency functions.

Table 6.2: Objective values of solutions of the instance shown in Figure 6-1

	maximum latency	average latency	unfairness
min-max flow	3/2	3/2	1
system optimum	$\{14/9, 20/9\}$	4/3	$\{7/4, 5/2\}$
user equilibrium	2	2	1

6.3 Computational Complexity

It follows from the work of Köhler and Skutella (2002) on the quickest s-t-flow problem with load-dependent transit times that the maximum latency problem considered here is NP-hard (though not necessarily in NP) when latencies include arbitrary nonlinear functions or when there are explicit arc capacities. Lemma 6.1 below implies that the general maximum latency problem is in NP, while Theorem 6.3 establishes its NP-hardness, even in the case of linear latencies and a single source and a single sink.

Note that the following result does not follow from ordinary flow decomposition as it is not clear how to convert a flow on arcs into a path flow such that the latency of the resulting paths remains bounded; in fact, it is a consequence of Theorem 6.3 that the latter problem is NP-hard, too.

Lemma 6.1. *Let x be a feasible flow for a multicommodity flow network with nonnegative latency functions. (Note that we do not need to assume that latencies are nondecreasing or continuous for this result.) Then, there exists another feasible flow x' such that $L(x') \leqslant L(x)$, and x' uses at most $|A|$ paths for each source-sink pair.*

Proof. Consider an arbitrary commodity $k \in K$. Let $P_1, \ldots, P_r$ be s_k-t_k-paths such that $x_{P_i} > 0$ for $i = 1, \ldots, r$, and $\sum_{i=1}^{r} x_{P_i} = d_k$. Slightly overloading notation, we let $P_1, \ldots, P_r$ also denote the arc incidence vectors of these paths. Let's assume that $r > |A|$. (Otherwise we are done.) Hence, the vectors $P_1, \ldots, P_r$ are linearly dependent and $\sum_{i=1}^{r} \lambda_i P_i = 0$ has a nonzero solution. Let's assume without loss of generality that $\lambda_r \neq 0$. We define a new flow x'' (not necessarily feasible) by setting $x''_{P_i} \overset{\text{def}}{=} x_{P_i} - \frac{\lambda_i}{\lambda_r} x_{P_r}$ for $i = 1, \ldots, r$, and $x''_P \overset{\text{def}}{=} x_P$ for all other paths P. Notice that under x'', the flow on

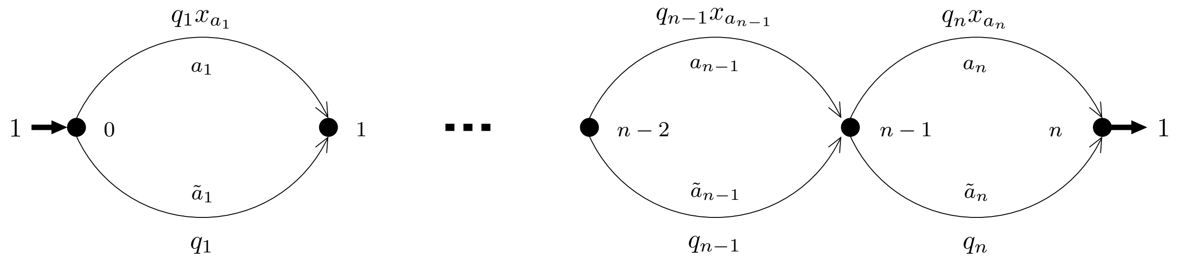


Figure 6-2: Instance used in the reduction from Partition

arcs does not change:

$$\sum_{i=1}^{r} P_i x''_{P_i} = \sum_{i=1}^{r-1} P_i x_{P_i} - \sum_{i=1}^{r-1} \frac{\lambda_i}{\lambda_r} P_i x_{P_r} = \sum_{i=1}^{r} P_i x_{P_i} \, .$$

Here, we used the linear dependency for the last equality. In particular, $L(x'') \leqslant L(x)$. Let us consider a convex combination x' of x and x'' that is nonnegative and uses fewer paths than x. Note that such a flow always exists because $x''_{P_r} = 0$, and the flow on some other paths $P_1, \ldots, P_{r-1}$ might be negative. Moreover, $L(x') \leqslant L(x)$, too. If x' still uses more than $|A|$ paths between s_k and t_k, we can iterate this process so long as necessary to prove the claim. □

Corollary 6.2. *The decision version of the maximum latency problem is in NP.*

Proof. Lemma 6.1 shows the existence of a succinct certificate. Indeed, there is a min-max flow using no more than $|K| \cdot |A|$ paths. □

We are now ready to prove that the maximum latency problem is in fact NP-hard. We present a reduction from Partition:

> Given: A set of n positive integer numbers $q_1, \ldots, q_n$.
>
> Question: Is there a subset $I \subset \{1, \ldots, n\}$ such that $\sum_{i \in I} q_i = \sum_{i \notin I} q_i$?

Theorem 6.3. *The decision version of the maximum latency problem is NP-complete, even when all latencies are linear functions and the network has a single OD pair.*

Proof. Given an instance of PARTITION, we define an instance of the maximum latency problem as depicted in Figure 6-2. The network consists of nodes $0, 1, \ldots, n$ with 0 representing the source and n the sink, and there is a unit demand. For $i = 1, \ldots, n$, the nodes $i - 1$ and i are connected with two arcs, namely a_i with latency $\ell_{a_i}(x_{a_i}) = q_i \, x_{a_i}$ and $\tilde{a}_i$ with latency $\ell_{\tilde{a}_i}(x_{\tilde{a}_i}) = q_i$.

Let $T \overset{\text{def}}{=} \frac{3}{4} \sum_{i=1}^{n} q_i$. Notice that the system optimum f^* has cost equal to T and $f_a^* = 1/2$ for all $a \in A$. We claim that the given instance of PARTITION is a YES-instance if and only if there is a solution to the maximum latency problem of maximum latency equal to T. Indeed, if there is a partition I, the flow that routes half a unit of flow along the 0-n-path composed of arcs a_i, $i \in I$, and $\tilde{a}_i$, $i \notin I$, and the other half along the complementary path has maximum latency T.

To prove the other direction, assume that we have a flow x of maximum latency equal to T. Therefore, $\overline{C}(x) \leqslant T$ (there is a unit demand), which implies that $\overline{C}(x) = T$ (it cannot be better than the optimal solution). As the arc flows of a system optimum are unique, this implies that $x_a = 1/2$ for all $a \in A$. Take any path P such that $x_P > 0$, and partition its arcs such that I contains the indices of the arcs $a_i \in P$. Then, $\frac{3}{4} \sum_{i=1}^{n} q_i = T = \ell_P(x) = \sum_{i \in I} \frac{q_i}{2} + \sum_{i \notin I} q_i$, and subtracting the left-hand side from the right-hand side yields $\sum_{i \in I} \frac{q_i}{4} = \sum_{i \notin I} \frac{q_i}{4}$. □

The following corollary states that a flow decomposition that achieves small path-lengths is difficult to compute.

Corollary 6.4. *Let x be a (path) flow in an s-t-network with linear latencies. Let $(x_a)_{a \in A}$ be the associated flow on arcs. Given just $(x_a)_{a \in A}$ and $L(x)$, it is NP-hard to compute a decomposition of this arc flow into a (path) flow x' such that $L(x') \leqslant L(x)$. In particular, it is NP-hard to recover a min-max flow even though its arc values are given.*

Recall that, as we discussed in Chapter 2, Corollary 6.4 neither holds for the system optimum nor the user equilibrium. In both cases any flow derived from an ordinary flow decomposition is indeed an optimal flow or an equilibrium flow, respectively. Nevertheless, an arbitrary decomposition of a system-optimal flow need not be good with respect to the maximum latency objective. To see that, consider the instance described in the proof of Theorem 6.3. The flow on paths that routes 1/2 along the path $a_1, a_2, \ldots, a_n$ and 1/2 along the path $\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_n$ is indeed a system optimum, but its maximum latency is $\sum_{i=1}^n q_i$ as opposed to the optimal solution which has value $\frac{3}{4}\sum_{i=1}^n q_i$. In Section 6.5, we prove a tight worst-case bound for the maximum latency of a system optimum.

Let us finally mention that Theorem 4.3 in Köhler and Skutella (2002) implies that the maximum latency problem is APX-hard when latencies can be arbitrary nonlinear functions.

6.4 Fairness

As explained in Section 2.3, user equilibria are fair. The next result establishes the same property for min-max s-t-flows in the case of linear latencies. Namely, a fair min-max flow always exists. Therefore, the difference between a user equilibrium and a min-max flow is that the latter may leave paths unused that are shorter than the ones carrying flow, a situation that cannot happen at equilibrium. The following result is not true for nonlinear latencies, as we shall see later.

Theorem 6.5. *Every instance of the single-source single-sink maximum latency problem with linear latency functions has an optimal solution that is fair.*

Proof. Consider an instance with demand d and latency functions $\ell_a(x_a) = q_a x_a + r_a$, for $a \in A$. Among all min-max flows, let $\hat{f}$ be the one that uses the smallest number of paths. Let $P_1, P_2, \ldots, P_u$ be these paths. Consider the following linear program:

$$\min \quad z \tag{6.1a}$$

$$\text{s.t.} \quad \sum_{a \in P_i} q_a x_a + r_a \leqslant z \qquad i = 1, \ldots, u, \tag{6.1b}$$

$$\sum_{P_h \ni a} x_{P_h} = x_a \qquad a \in A, \tag{6.1c}$$

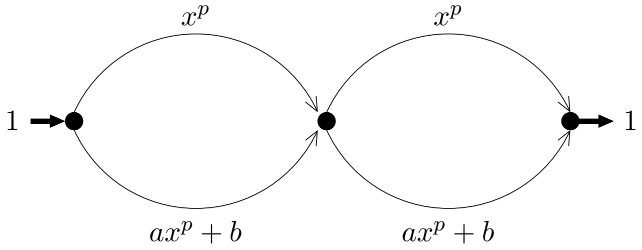


Figure 6-3: Instance with nonlinear latencies that illustrates that fair min-max flows may not exist

$$\sum_{i=1}^{u} x_{P_i} = d \tag{6.1d}$$

$$x_{P_i} \geqslant 0 \qquad\qquad i = 1, \ldots, u. \tag{6.1e}$$

Note that this linear program has $u + 1$ variables. Furthermore, by construction, it has a feasible solution with $z = L(\hat{f})$, and there is no solution with $z < L(\hat{f})$. Therefore, an optimal basic feasible solution gives a min-max flow that satisfies with equality u of the inequalities (6.1b) and (6.1e). As $x_{P_i} > 0$ for all i because of the minimality assumption, all inequalities (6.1b) have to be tight. □

A byproduct of this proof is that an arbitrary flow can be transformed into a fair one without increasing its maximum latency. In fact, just solve the corresponding linear program. An optimal basic feasible solution will either be fair or it will use fewer paths. In the latter case, eliminate all paths carrying zero flow and repeat until a fair solution is found.

Notice that the min-max flow may not be fair for nonlinear functions. A modification of the example presented in Section 6.2 features high unfairness with latencies that are polynomials of degree p, for $p \geqslant 2$ (see Figure 6-3).

When $a = (1 + \varepsilon)^{p-1}$ and $b = 2 - \left(\frac{1+\varepsilon}{2+\varepsilon}\right)^{p-1} - \delta$ for some $\varepsilon > 0$ and $\delta > 0$ such that $b > 1$, the min-max flow routes $\frac{1}{2+\varepsilon}$ units of flow along the "top-bottom" and "bottom-top" paths, respectively, and $\frac{\varepsilon}{2+\varepsilon}$ units of flow along the "top-top" path. It is not hard to see that this flow is optimal. Indeed, the "bottom-bottom" path is too long to carry any flow. Moreover, by symmetry, the "top-bottom" and "bottom-top" paths have to carry

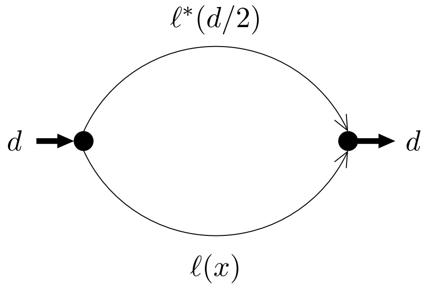


Figure 6-4: Instance showing that Theorem 5.12 is tight

the same amount of flow. Letting the variable x denote the flow on the "top-top" path, the flow on both top arcs is $\frac{1+x}{2}$, and that of both bottom arcs is $\frac{1-x}{2}$. Summing along paths, we get that the latency of the "top-top" path is $2\left(\frac{1+x}{2}\right)^p$, which is always smaller than that of the other two paths, which is $\left(\frac{1+x}{2}\right)^p + a\left(\frac{1-x}{2}\right)^p + b$. Finally, we compute the optimal solution of $\min\left\{\left(\frac{1+x}{2}\right)^p + a\left(\frac{1-x}{2}\right)^p + b : 0 \leqslant x < 1\right\}$ and get $\hat{f}_{\text{top-top}} = \frac{\varepsilon}{2+\varepsilon}$, as specified before.

Let us compute the unfairness of this solution. The "top-top" path has latency equal to $2\left(\frac{1+\varepsilon}{2+\varepsilon}\right)^p$, which tends to $\left(\frac{1}{2}\right)^{p-1}$ as $\varepsilon \to 0$. The latency of the other two paths used by the optimum is equal to $2-\delta$. Therefore, the unfairness of this min-max flow is arbitrarily close to 2^p.

As explained earlier, system optima may route flow along paths that users would not choose. Recall that the modified latency function $\ell^*_a(x_a)$ was defined as $(\ell_a(x_a)x_a)' = \ell_a(x_a) + \ell'_a(x_a)x_a$. In Chapter 5, we proved that when latencies are drawn from $\mathcal{L}$, the unfairness of a system optimal flow is bounded by $\gamma(\mathcal{L})$, where $\gamma(\mathcal{L})$ was defined as the smallest value that satisfies $\ell^*_a(x) \leqslant \gamma(\mathcal{L})\ell_a(x)$ for all $\ell \in \mathcal{L}$ and all $x \geqslant 0$. Here, we prove that the bound shown in Theorem 5.12 turns out to be tight.

To see that consider the instance displayed in Figure 6-4. It is easy to see that the system optimum routes half of the demand along each arc, implying that the unfairness is $\ell^*(d/2)/\ell(d/2)$. Taking the supremum of that ratio over $d \geqslant 0$ and $\ell \in \mathcal{L}$ yields $\gamma(\mathcal{L})$. Moreover, note that this example is also tight if constrained system optima are considered (see Corollary 5.14).

6.5 Price of Anarchy and Related Approximation Results

As discussed earlier, Nash equilibria in general and user equilibria in particular are known to be inefficient (Section 2.5). It is quite appealing that in the routing game considered here, the price of anarchy with respect to the maximum latency objective is small too. It was first noted by Weitz (2001) that in networks with only one source and one sink, any upper bound on the price of anarchy for the total latency is an upper bound on the price of anarchy for the maximum latency. We present a multicommodity version of this result that relies on the findings for the total latency objective shown by Theorems 2.8 and 3.10. Recall that $L_k(x)$ denotes the maximum latency incurred by a flow-carrying path in $\mathcal{P}_k$ under a flow x, and d_k denotes the demand rate corresponding to OD pair k.

Theorem 6.6. *Consider a set $\mathcal{L}$ of continuous and nondecreasing latency functions, and a multicommodity flow network with latency functions drawn from $\mathcal{L}$. Let f be a user equilibrium and $\hat{f}$ be a min-max flow. For each commodity $k \in K$, $L_k(f) \leqslant \frac{d}{d_k}\alpha(\mathcal{L})L(\hat{f})$, where $d \overset{def}{=} \sum_{k\in K} d_k$ is the total demand.*

Proof. Let f^* be the system optimum. Then,

$$d_k L_k(f) \leqslant d\,\overline{C}(f) \leqslant d\,\alpha(\mathcal{L})\overline{C}(f^*) \leqslant d\,\alpha(\mathcal{L})\overline{C}(\hat{f}) \leqslant d\,\alpha(\mathcal{L})L(\hat{f})\,.$$

Here, the first inequality holds because f is a user equilibrium, the second inequality is that of Theorem 3.10,[1] the third one comes from the optimality of f^*, and the last one just expresses that the average latency cannot be larger than the maximum latency. □

The proof of Theorem 6.6 implies that if for a given single-source single-sink instance an equilibrium flow f happens to be a system optimum, then f is also optimal for

[1] Although we do not need side constraints for this result, we rely on this theorem instead of Theorem 2.8 to be able to relax the assumptions on latency functions.

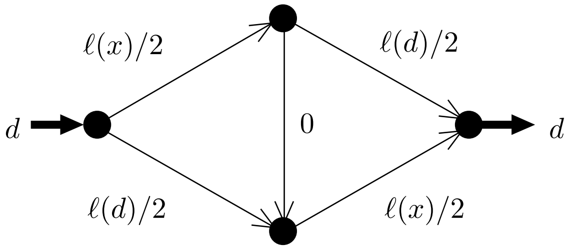


Figure 6-5: Instance showing that Theorem 6.6 is tight for single-commodity networks

the maximum latency objective.[2] For single-source single-sink networks with general latency functions, Theorem 6.6 shows that computing a user equilibrium is an $\alpha(\mathcal{L})$-approximation algorithm for the maximum latency problem. Notice that this guarantee is tight as shown by the example given in Figure 6-5, which generalizes the Braess' Paradox network described in Section 2.4.2. Indeed, the latency of a user equilibrium is $\ell(d)$ while the maximum latency of a min-max flow, which coincides with the system optimum, is

$$\ell(d) - \max_{0 \leqslant x \leqslant d} \left\{ \frac{x}{d} \big(\ell(d) - \ell(x) \big) \right\} .$$

As in Section 3.6, taking the supremum over $d \geqslant 0$ and $\ell \in \mathcal{L}$, the ratio of the latency of the user equilibrium to that of the min-max flow is arbitrarily close to $\alpha(\mathcal{L})$.

For instances with multiple sources and a single sink, the maximum latency of a user equilibrium is unbounded with respect to that of an optimal solution, even with linear latencies. In fact, we will show that the price of anarchy cannot be better than $\Omega(n)$, where n is the number of nodes in the network. Weitz (2001) showed that the price of anarchy is unbounded in the case of two commodities, and Roughgarden (2004b) proved that it is at most $n-1$ if there is a common source and sink.

Theorem 6.7. *The price of anarchy (with respect to the maximum latency) in a single-commodity network with multiple sources and a single sink is $\Omega(n)$, even if all latencies are linear functions.*

Proof. Fix a constant $\varepsilon > 0$ and consider the instance presented in Figure 6-6. Nodes

[2] Recall from Chapter 2 that the system optimum and the user equilibrium coincide when latencies are monomials of a common degree.

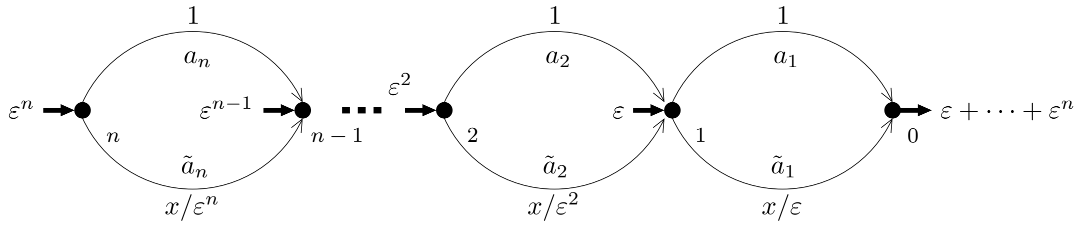


Figure 6-6: Instance showing that user equilibria can be arbitrarily bad with respect to the maximum latency objective when multiple sources are present

$n, n-1, \ldots, 1$ are the sources while node 0 is the sink. Nodes i and $i-1$ are connected with two arcs: a_i with constant latency equal to 1 and $\tilde{a}_i$ with latency equal to x/ε^i. Let the demand entering node $i > 0$ be ε^i. The user equilibrium of this instance routes the flow along paths of the form $\tilde{a}_i, a_{i-1}, \ldots, a_1$ and has maximum latency n. To show the claim, it suffices to exhibit a good solution. For instance, for origin i, let its demand flow along the path $a_i, \tilde{a}_{i-1}, \ldots, \tilde{a}_1$. Under this flow, the load of $\tilde{a}_i$ is equal to $\varepsilon^{i+1} + \cdots + \varepsilon^n$ and its traversal time is $(\varepsilon^{i+1} + \cdots + \varepsilon^n)/\varepsilon^i = \varepsilon^1 + \cdots + \varepsilon^{n-i}$. Hence, we can bound the maximum latency from above by $1 + \frac{n\varepsilon}{1-\varepsilon}$, which tends to 1 when $\varepsilon \to 0$. □

In contrast to our results of Chapter 3 about the total latency objective, the previous theorem implies that equilibria are bad in single-source single-sink networks with capacities.

Corollary 6.8. *Consider a single-source single-sink network with explicit arc capacities. With respect to the maximum latency, the worst-case coordination ratio of the best user equilibria is unbounded, even if all latencies are linear functions.*

Proof. It suffices to show an instance with high coordination ratio. Starting from that of Theorem 6.7, we create one with capacities using a common network transformation (Ahuja, Magnanti, and Orlin 1993). Indeed, we add a super-source $n+1$, and for each node $i = 1, \ldots, n$, we connect $n+1$ to i with an arc of capacity equal to the demand of OD pair $(i, 0)$, and with latency function $\ell_{(i,0)}(x) \overset{\text{def}}{=} 0$. Finally, we let the demand of the new instance be the single OD pair $(n+1, 0)$ with rate equal to the total demand of the original network.

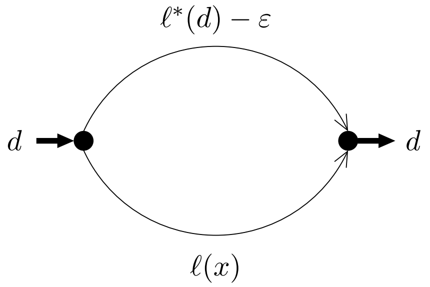


Figure 6-7: Instance showing that Theorem 6.9 is tight

As all feasible flows in the new instance saturate every arc $(n+1, i)$ for $i = 1, \ldots, n$, flows in the original and the new instances are in one-to-one correspondence. Therefore, the extension of a min-max flow is a min-max flow. More interestingly, the extension of a user equilibrium is a user equilibrium with side constraints, according to Definition 3.3. Indeed, users of an OD pair $(k, 0)$ cannot switch to another OD pair $(k', 0)$ because arc $(n+1, k')$ is saturated. As the flow was at equilibrium in the original instance, users do not have an incentive to switch to another path starting with arc $(n+1, k)$. The original instance had an essentially unique equilibrium (recall that this means that if there is more than one, they all share the same objective value), and so does the new one. Hence, the coordination ratio is that of the original instance, which was shown to be large. □

Of course, the price of anarchy is unbounded because it refers to an arbitrary equilibrium. Let us also remark that for the instance described in the previous corollary, the extension of *any* flow of the original instance is a user equilibrium with capacities (Definition 3.1).

Going back to the approximations guarantees, let us note that in the single-source single-sink case, user equilibria are not the only good approximations to the maximum latency problem. For instance, Corollary 5.13 implies that system optima are also close to optimality with respect to the maximum latency objective.

Theorem 6.9. *Let $\mathcal{L}$ be a family of differentiable and nondecreasing latency functions. For single-source single-sink instances with latency functions drawn from $\mathcal{L}$, computing a system optimum is a $\gamma(\mathcal{L})$-approximation algorithm for the maximum latency problem.*

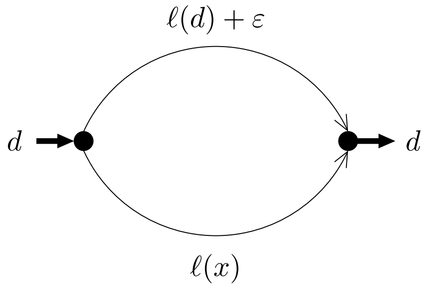


Figure 6-8: Instance showing that Theorem 6.10 is tight

Proof. In Corollary 5.13, replace x by a min-max flow. □

The bound given in Theorem 6.9 is best possible. To see this, consider the instance depicted in Figure 6-7. The min-max flow routes the entire demand along the lower arc, for a small enough $\varepsilon > 0$. On the other hand, the unique system optimum has to satisfy $\ell^*(f^*) = \ell^*(d) - \varepsilon$, where f^* is the flow along the lower arc. Therefore, the upper arc has positive flow and the maximum latency is $\ell^*(d) - \varepsilon$. The ratio between the maximum latencies of the two solutions is arbitrarily close to $\ell^*(d)/\ell(d)$. Taking the supremum over $d \geqslant 0$ and $\ell \in \mathcal{L}$ shows that the bound in Theorem 6.9 is tight.

To complete Table 6.1, let us prove that the average latency of the min-max flow is not too far from that of the system optimum.

Theorem 6.10. *Consider a set $\mathcal{L}$ of continuous and nondecreasing latency functions. Let $\hat{f}$ be a min-max flow and f^* be a system optimum for an instance with a single source, a single sink, and latencies drawn from $\mathcal{L}$. Then, $\overline{C}(\hat{f}) \leqslant \alpha(\mathcal{L})\overline{C}(f^*)$.*

Proof. Note that $\overline{C}(\hat{f}) \leqslant L(\hat{f}) \leqslant L(f) = \overline{C}(f) \leqslant \alpha(\mathcal{L})\overline{C}(f^*)$, where f is the user equilibrium of the instance. □

Again, the guarantee given in the previous theorem is tight. To show this, it is enough to note that the equilibrium flow and the min-max flow coincide in the example of Figure 6-8, and their average latency is $\ell(d)$. Moreover, the average latency of the system optimum is arbitrary close to

$$\ell(d) - \max_{0 \leqslant x \leqslant d} \left\{ \frac{x}{d} \left(\ell(d) - \ell(x)\right) \right\} .$$

Table 6.3: Overview of approximation guarantees for single-source single-sink networks when latencies belong to a given set $\mathcal{L}$. All bounds are tight. The "?" indicates that no upper bound is known; recall from the example depicted in Figure 6-3 that 2^p is a lower bound for polynomials of degree p, for $p \geqslant 2$.

	maximum latency	average latency	unfairness
min-max flow	1	$\alpha(\mathcal{L})$	?
system optimum	$\gamma(\mathcal{L})$	1	$\gamma(\mathcal{L})$
user equilibrium	$\alpha(\mathcal{L})$	$\alpha(\mathcal{L})$	1

Taking the supremum of the ratio of these two values over $d \geqslant 0$ and $\ell \in \mathcal{L}$ completes the argument.

Let us finally remark that, for other traffic games, there always exist flows that simultaneously minimize the average and the maximum travel time (Fotakis, Kontogiannis, and Spirakis 2004). In other words, the bounds corresponding to the previous two results would be 1.

6.6 Discussion

We have shown that computing a flow of minimum maximum latency is NP-hard, even in the single-source single-sink case and with linear latency functions. Still, the problem admits a solution that is fair. We have proved tight bounds between the different solutions and with respect to different objectives. For instance, we have shown that two standard solutions in networks problems give constant factor approximations to this problem. Indeed, on the one hand, the coordination ratio of user equilibria is $\alpha(\mathcal{L})$ and they are fair. On the other, the ratio of the maximum latency of system optima to that of the min-max flow is $\gamma(\mathcal{L})$, while their unfairness is also bounded by $\gamma(\mathcal{L})$. In Table 6.3, we summarize the findings for single-source single-sink networks with latencies drawn from a given class $\mathcal{L}$ of allowable latency functions.

Let us finally note that a fourth objective could have been considered. We refer to a different generalization of the problem of Koutsoupias and Papadimitriou (1999). They considered the maximum latency in a network in which all paths consist of just a single arc. Instead of generalizing that objective to the maximum latency of a path, we could as well consider the problem of minimizing the maximum latency of the arcs, i.e., minimizing the length of the *bottleneck* arc, defined as the arc with maximum latency among those with flow. This objective would be of interest to telecommunication network's service providers because they try to minimize the arc loads so that, in an event of an arc failure, sufficient capacity is available for alternative routes.

In contrast to the maximum latency problem, this new problem can be solved in polynomial time, if latency functions are convex. To see that, we can "guess" the optimal solution. Obviously, arcs for which $\ell_a(0)$ is larger than our guess cannot be used in an optimal solution. We, therefore, let $\tilde{A}$ include the arcs of A that are shorter than the guess. We formulate the following convex program:

$$\begin{aligned}
\min \quad & z && && \text{(6.2a)}\\
\text{s.t.} \quad & \ell_a(x_a) \leqslant z && a \in \tilde{A}, && \text{(6.2b)}\\
& \sum_{P \ni a} x_P = x_a && a \in \tilde{A}, && \text{(6.2c)}\\
& \sum_{P \in \mathcal{P}_k} x_P = d_k && k \in K, && \text{(6.2d)}\\
& x_P \geqslant 0 && P \in \mathcal{P}, && \text{(6.2e)}
\end{aligned}$$

where $\mathcal{P}$ is modified to include paths consisting of arcs in $\tilde{A}$ only. Consider an optimal solution to the previous problem. If all the inequalities (6.2b) corresponding to arcs that carry flow are not tight, an empty arc is preventing z to be smaller. Hence, our guess was incorrect. Updating the guess with the maximum latency among arcs with flow, and repeating as many times as needed, solves the problem in polynomial time.

In terms of approximation, the optimal solution to this problem can be arbitrarily inefficient with respect to the other objectives we considered in this chapter and vice-versa. Let us call the optimal solution with respect to the new objective the "bottleneck flow." Consider an instance with two nodes connected with two paths that has to route

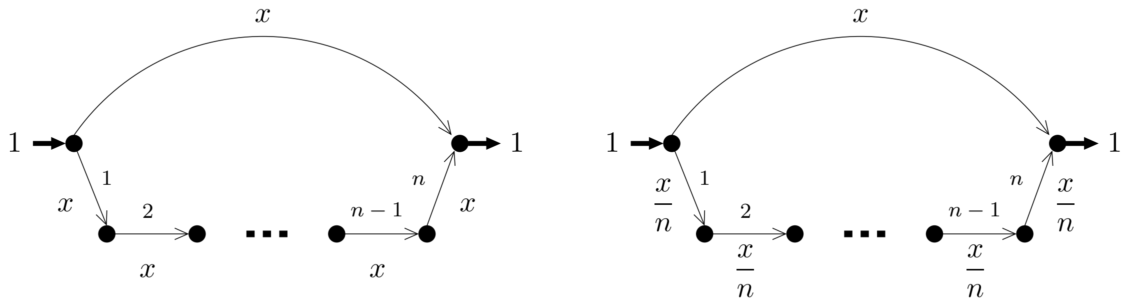


Figure 6-9: Examples for the bottleneck objective

a unit demand (Figure 6-9, *left*). The first path consists of a single arc and the second is given by a chain of n arcs; the latency of each arc is equal to its flow. The bottleneck flow is 1/2 along all the arcs, and the system optimum, the user equilibrium and the min-max flow coincide and are equal to $\frac{1}{n+1}$ along the chain and $\frac{n}{n+1}$ along the other arc. This shows that the bottleneck flow does arbitrarily bad with respect to the other three objectives.

Conversely, consider the same instance but now set the latencies of the arcs in the chain to $\frac{x}{n}$, where x is the flow on the arc (Figure 6-9, *right*). Now, the bottleneck flow is $\frac{n}{n+1}$ along the arcs on the chain and $\frac{1}{n+1}$ on the other arc. In this case, the system optimum, the user equilibrium and the min-max flow coincide, too, but are equal to 1/2 everywhere. This shows that the optimal solutions with respect to the three objectives are arbitrarily bad with respect to the bottleneck objective.

Chapter 7

Conclusion

One of the main goals of this dissertation has been to understand the consequences of a *laisez-faire* approach in systems where central coordination is impractical. In other words, we have considered the following question: How far from optimality is the system if no central coordination is imposed? This has been a long standing open question. For example, Mahmassani and Peeta (1993) wrote:

> ... the extent of the differences between SO [system optimum] and UE [user equilibrium] solutions, particularly in terms of overall system cost, is not known. This is very important for ATIS [Advanced Traveler Information Systems] because if the two solutions are not perceptibly different, coordinated cooperative SO route guidance imposed by a central controller may not be necessary, and descriptive information that is less complicated and simpler to disseminate to noncooperating drivers may be sufficient.
>
> [...]
>
> ... although mathematical relationships among traffic flow variables are reasonably well established for arterials and intersections, the intricacies of interactions at the network level preclude analytic derivability of network-wide traffic relationships from the link-level traffic models.

Nevertheless, this question has recently been answered for a certain traffic model (Roughgarden and Tardos 2002; Roughgarden 2003b), and our work contributes to addressing it by providing a positive answer for a more general model that is commonly used in transportation networks.

The proper use of route guidance devices with the objective of improving the utilization of road networks by giving more information to drivers has been one of the most active research areas in traffic engineering. Indeed, the ultimate goal of Intelligent Transportation Systems is making the actual traffic close to the system optimum. Yet, not all drivers would have the incentive to follow a corresponding route recommendation under system-optimal routing; actually, some would face rather long "detours." Therefore, most of the recent approaches in transportation science are content with computing a user equilibrium. That is, users are guided onto paths they would—in theory—take anyway. Our results give an a posteriori justification for doing so; in fact, we have shown for a broader class of networks than considered before that the expense of working with user equilibria instead of system optima is limited.

The introduction of side constraints proposed in Chapter 3 gives rise to multiple equilibria. In particular, the price of anarchy has been shown to be unbounded, even in the case of linear latency functions. Nevertheless, it is reassuring that the best user equilibrium is still close to the system optimum, despite the presence of side constraints. Even though that equilibrium happens to be difficult to compute, an equilibrium of good quality, namely the Beckmann user equilibrium, can be computed efficiently. Although Beckmann user equilibria are not appropriate for describing user behavior because it is not possible to predict which of the many equilibria users will adhere to,[1] Beckmann user equilibria can be used for prescribing user behavior. In fact, if the system optimal solution with user constraints that we have described in Chapters 4 and 5 cannot be implemented, a more conservative choice would be an equilibrium that is efficient and easily computable; that is precisely the Beckmann user equilibrium.

With respect to route guidance, the ability to guide people efficiently has potentially a significant impact. On the one hand, recall that the study conducted by the Texas

[1] Unless Observation 3.7 is used to argue that, behaviorally, the Beckmann user equilibrium is a "natural" equilibrium and therefore it is *the* equilibrium to be expected. As that argument requires a specific interpretation of the meaning of side constraints and we have not conducted experiments to determine the validity of that interpretation, we prefer to not make such a claim.

Transportation Institute (2002) estimated that congestion costs the United States $67.5 billion dollars per year. That implies that huge nationwide savings can be expected from even a small percentual reduction in congestion. On the other hand, besides minimizing the total travel time, our route guidance system also computes solutions with small travel time for each of the individual users (user equilibria, although fair, may be bad for all users as exemplified by the Braess Paradox).

Finally, we have also considered alternative objective functions relevant to other application domains. The price of anarchy computed earlier in Chapter 3 is robust against changes in the objective function. For s-t-networks, the price of anarchy does not change if the maximum latency objective is used to evaluate the solution performance. Furthermore, system optima, user equilibria, and min-max flows are all close to each other when evaluated with respect to the three objectives we have considered in the dissertation (total latency, unfairness, and maximum latency).

Before concluding with open questions, let us note that although we have exclusively worked with network games, there are other research efforts related to the price of anarchy that consider other domains. Vetta (2002) proposed a general framework that fits multiple applications. In his model, an instance is given by a ground set and players have to choose a subset of those elements. Assuming that the cost function is submodular,[2] the author proved an upper bound for the price of anarchy. He applied that result to competitive facility location problems, auctions, and a road traffic model similar to that of Chapter 2. Johari and Tsitsiklis (2004) analyzed resource allocation games. In particular, they considered a telecommunication system in which individuals bid for capacity according to private utilities. That article extends the work of Kelly (1997) on rate control to the case in which users are price anticipating instead of price takers. The authors proved a tight bound for the price of anarchy for this model (which surprisingly is again 4/3, but as opposed to our model, linear utilities are the worst possible). Acemoglu and Ozdaglar (2003) analyzed a telecommunication network in which a (monopolist) service provider sets prices for its links. Users, in a second stage, selfishly select the amount of

[2] Basically, submodular functions correspond to those with decreasing marginal utility, and therefore are common in several applications.

flow to transmit and their routes. They proved the existence of a unique equilibrium of this two-stage game. Interestingly, this equilibrium achieves full efficiency for the routing problem. Finally, Perakis (2004) argued that her results on the price of anarchy for non-separable latencies (described in Section 3.9) can be used to guarantee the efficiency of an equilibrium in a competitive multi-period pricing problem.

The concept of the price of anarchy has been gaining momentum lately, and there now exist many research groups working on these problems. In the author's opinion, this area of research is particularly exciting, not only because of the interesting questions that exist, but also because the concepts and tools it requires have been drawn from different application domains. These concepts and tools have facilitated some elegant results and proofs.

Bibliography

Aashtiani, H. Z. and T. L. Magnanti (1981). Equilibria on a congested transportation network. *Siam Journal on Algebraic and Discrete Methods 2*, 213–226.

Acemoglu, D. and A. Ozdaglar (2003). Flow control, routing, and performance from service provider viewpoint. LIDS Report WP-1696, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA.

Ahuja, R. K., T. L. Magnanti, and J. B. Orlin (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs, NJ.

Aneja, Y. P., V. Aggarwal, and K. P. K. Nair (1983). Shortest chain subject to side constraints. *Networks 13*, 295–302.

Aneja, Y. P. and K. P. K. Nair (1978). The constrained shortest path problem. *Naval Research Logistics Quarterly 25*, 549–555.

Anshelevich, E., A. Desgupta, É. Tardos, and T. Wexler (2003). Near-optimal network design with selfish agents. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, San Diego, CA, pp. 511–520. ACM Press, New York, NY.

Arezki, Y. and D. Van Vliet (1990). A full analytical implementation of the Partan/Frank-Wolfe algorithm for equilibrium assignment. *Transportation Science 24*, 58–62.

Arnott, R. and K. Small (1994). The economics of traffic congestion. *American Scientist 82*, 446–455.

Aronson, J. E. (1989). A survey of dynamic network flows. *Annals of Operations Research 20*, 1–66.

Bar-Gera, H. (2002). Transportation network test problems. `http://www.bgu.ac.il/~bargera/tntp/`.

Beccaria, G. and A. Bolelli (1991). Modelling and assessment of dynamic route guidance: The MARGOT project. In *Proceedings of the IEEE Vehicle Navigation & Information Systems Conference*, Dearborn, MI, Volume 1, pp. 117–126. Society of Automotive Engineers, Warrendale, PA.

Beckmann, M. J., C. B. McGuire, and C. B. Winsten (1956). *Studies in the Economics of Transportation*. Yale University Press, New Haven, CT. Available at `http://www.rand.org/publications/RM/RM1488.pdf`.

Ben-Akiva, M. E. (1985). Dynamic network equilibrium research. *Transportation Research, Part A 19A*, 429–431.

Ben-Akiva, M. E., M. Bierlaire, J. Bottom, H. N. Koutsopoulos, and R. G. Mishalani (1997). Development of a route guidance generation system for real-time application. In M. Papageorgiou and A. Pouliezos (Eds.), *Proceedings of the 8th IFAC Symposium on Transportation Systems*, Chania, Greece, pp. 405–410. Elsevier Science, Oxford.

Ben-Akiva, M. E., E. Cascetta, and H. Gunn (1995). An on-line dynamic traffic prediction model for an inter-urban motorway network. In N. H. Gartner and G. Improta (Eds.), *Urban Traffic Networks. Dynamic Flow Modelling and Control*, pp. 83–122. Springer, Berlin.

Ben-Akiva, M. E., A. de Palma, and I. A. Kaysi (1996). The impact of predictive information on guidance efficiency: An analytical approach. In L. Bianco and P. Toth (Eds.), *Advanced Methods in Transportation Analysis: An Analytical Approach*, pp. 413–432. Springer, Berlin.

Berenbrink, P., L. A. Goldberg, P. Goldberg, and R. Martin (2003). Utilitarian resource assignment. Manuscript. Available at `http://www.dcs.warwick.ac.uk/~pwg/`.

Bergendorff, P., D. W. Hearn, and M. V. Ramana (1997). Congestion toll pricing of traffic networks. In P. M. Pardalos, D. W. Hearn, and W. W. Hager (Eds.), *Network Optimization*, Volume 450 of *Lecture Notes in Economics and Mathematical Systems*, pp. 51–71. Springer, Berlin.

Bernstein, D. and T. E. Smith (1994). Equilibria for networks with lower semicontinuous costs: With an application to congestion pricing. *Transportation Science 28*, 221–235.

Bertsekas, D. P. (1998). *Network Optimization: Continuous and Discrete Models.* Athena Scientific, Belmont, MA.

Bertsekas, D. P. (1999). *Nonlinear Programming.* Athena Scientific, Belmont, MA.

Bertsekas, D. P. and R. Gallager (1991). *Data Networks.* Prentice-Hall, Englewood Cliffs, NJ.

Bottom, J. A. (2000). *Consistent Anticipatory Route Guidance.* Ph. D. thesis, Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Boyce, D. E. (1989). Contributions of transportation network modelling to the development of a real-time route guidance system. In D. Batten and R. Thord (Eds.), *Transportation for the Future*, Proceedings of the First International Conference on Transportation for the Future, Södertälje, Sweden, May 1988, pp. 161–177. Springer, Berlin.

Boyce, D. E., B. N. Janson, and R. W. Eash (1981). The effect on equilibrium trip assignment of different link congestion functions. *Transportation Research, Part A 15A*, 223–232.

Braess, D. (1968). Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung 12*, 258–268.

Branston, D. (1976). Link capacity functions: A review. *Transportation Research 10*, 223–236.

Bureau of Public Roads (1964). Traffic assignment manual. U.S. Department of Commerce, Urban Planning Division, Washington D.C.

Burkard, R. E., K. Dlaska, and B. Klinz (1993). The quickest flow problem. *ZOR Methods and Models of Operations Research 37*, 31–58.

Chalmet, L., R. Francis, and P. Saunders (1982). Network models for building evacuation. *Management Science 28*, 86–106.

Charnes, A. and W. W. Cooper (1961). Multicopy traffic network models. In R. Herman (Ed.), *Theory of Traffic Flow*, Proceedings of the Symposium on the Theory of Traffic Flow held at the General Motors Research Laboratories, Warren, MI, 1959, pp. 85–96. Elsevier, Amsterdam.

Chau, C. K. and K. M. Sim (2003). The price ofanarchy for non-atomic congestion games with symmetric cost maps and elastic demands. *Operations Research Letters 31*, 327–334.

Chen, K. and S. E. Underwood (1991). Research on anticipatory route guidance. In *Proceedings of the IEEE Vehicle Navigation & Information Systems Conference*, Dearborn, MI, Volume 1, pp. 551–556. Society of Automotive Engineers, Warrendale, PA.

Chou, Y.-L., H. E. Romeijn, and R. L. Smith (1998). Approximating shortest paths in large-scale networks with an application to Intelligent Transportation Systems. *INFORMS Journal on Computing 10*, 163–179.

Climaco, J. C. N. and E. Q. V. Martins (1982). A bicriterion shortest path algorithm. *European Journal of Operational Research 11*, 399–404.

Cohen, J. E. and F. P. Kelly (1990). A paradox of congestion in a queuing network. *Journal of Applied Probability 27*, 730–734.

Cohen, S. (1991). Flow variables. In M. Papageorgiou (Ed.), *Concise Encyclopedia of Traffic & Transportation Systems*, pp. 139–143. Pergamon Press, Oxford.

Correa, J. R., A. S. Schulz, and N. E. Stier-Moses (2004a). Computational complexity, fairness, and the price of anarchy of the maximum latency problem. In G. Nemhauser and D. Bienstock (Eds.), *Proceedings of the 10th Conference on Integer Programming and Combinatorial Optimization (IPCO)*, New York, NY, Volume 3064 of *Lecture Notes in Computer Science*. Springer, Heidelberg. To Appear.

Correa, J. R., A. S. Schulz, and N. E. Stier-Moses (2004b). Selfish routing in capacitated networks. *Mathematics of Operations Research*. To appear. A preliminary version appeared as Working Paper No. 4319-03, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA.

Czumaj, A. (2004). Selfish routing on the Internet. In J. Leung (Ed.), *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*, Volume 1 of *Chapman & Hall/CRC Computer & Information Science Series*. CRC Press, Boca Raton, FL.

Czumaj, A. and B. Vöcking (2004). Tight bounds for worst-case equilibria. *Journal of Algorithms*. To appear. A preliminary version appeared in SODA 2002.

Dafermos, S. C. (1972). The traffic assignment problem for multiclass-user transportation networks. *Transportation Science 6*, 73–87.

Dafermos, S. C. (1980). Traffic equilibrium and variational inequalities. *Transportation Science 14*, 42–54.

Dafermos, S. C. and F. T. Sparrow (1969). The traffic assignment problem for a general network. *Journal of Research of the U.S. National Bureau of Standards 73B*, 91–118.

Daganzo, C. F. (1977a). On the traffic assignment problem with flow dependent costs—I. *Transportation Research 11*, 433–437.

Daganzo, C. F. (1977b). On the traffic assignment problem with flow dependent costs—II. *Transportation Research 11*, 439–441.

de Palma, A. and Y. Nesterov (1998). Optimization formulations and static equilibrium in congested transportation networks. CORE Discussion Paper 9861, Université Catholique de Louvain, Louvain-la-Neuve, Belgium.

Downs, A. (1962). The law of peak-hour expressway congestion. *Traffic Quarterly 16*, 393–409.

Dupuit, J. (1849). On tolls and transport charges. *Annales des ponts et chaussées*. Reprinted in *International Economic Papers 11* (1962), 7–31.

DynaMIT (2002). DynaMIT/DynaMIT-P (v 0.9) User's Manual. Intelligent Transportation Systems Program, Massachusetts Institute of Technology. Available at `http://www.dynamictrafficassignment.org/dmp_ug.pdf`.

Dynasmart (2002). Dynasmart-P (v 0.9) User's Guide. Center for Transportation Research, University of Texas at Austin. Available at `http://www.dynamictrafficassignment.org/dsp_ug.pdf`.

Feldmann, R., M. Gairing, T. Lücking, B. Monien, and M. Rode (2003a). Nashification and the coordination ratio for a selfish routing game. In J. C. M. Baeten, J. K. Lenstra, J. Parrow, and G. J. Woeginger (Eds.), *Proceedings of the 30th International Colloquium on Automata, Languages, and Programming (ICALP)*, Eindhoven, The Netherlands, Volume 2719 of *Lecture Notes in Computer Science*, pp. 514–526. Springer, Heidelberg.

Feldmann, R., M. Gairing, T. Lücking, B. Monien, and M. Rode (2003b). Selfish routing in non-cooperative networks: A survey. In B. Rovan and P. Vojtáš (Eds.), *Proceedings of the 28th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, Bratislava, Slovakia, Volume 2747 of *Lecture Notes in Computer Science*, pp. 21–45. Springer, Heidelberg.

Ferris, M. C. and A. Ruszczyński (1997). Robust path choice and vehicle guidance in networks with failures. Technical Report 97–04, Computer Sciences Department, University of Wisconsin.

Fleischer, L. and M. Skutella (2002). The quickest multicommodity flow problem. In W. J. Cook and A. S. Schulz (Eds.), *Proceedings of the 9th Conference on Integer Programming and Combinatorial Optimization (IPCO)*, Cambridge, MA, Volume 2337 of *Lecture Notes in Computer Science*, pp. 36–53. Springer, Heidelberg.

Fleischer, L. and M. Skutella (2003a). Minimum cost flows over time without intermediate storage. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Baltimore, MD, pp. 66–75. SIAM, Philadelphia, PA.

Fleischer, L. and M. Skutella (2003b). Quickest flows over time. Manuscript. Available at `http://www.mpi-sb.mpg.de/~skutella/FleischerSkutella.pdf`. Different parts of this work have appeared in preliminary form in Fleischer and Skutella (2002) and Fleischer and Skutella (2003a).

Florian, M. (1977). A traffic equilibrium model of travel by car and public transit modes. *Transportation Science 11*, 166–179.

Florian, M. (1986). Nonlinear cost network models in transportation analysis. *Mathematical Programming 26*, 167–196.

Florian, M., J. Guélat, and H. Spiess (1987). An efficient implementation of the "Partan" variant of the linear approximation method for the network equilibrium problem. *Networks 17*, 319–339.

Florian, M. and D. W. Hearn (1995). Network equilibrium models and algorithms. In M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser (Eds.), *Network Routing*, Volume 8 of *Handbooks in Operations Research and Management Science*, Chapter 6, pp. 485–550. Elsevier, New York, NY.

Ford, L. R. and D. R. Fulkerson (1958). Constructing maximal dynamic flows from static flows. *Operations Research 6*, 419–433.

Fotakis, D., S. Kontogiannis, E. Koutsoupias, M. Mavronicolas, and P. Spirakis (2002). The structure and complexity of Nash equilibria for a selfish routing game. In P. Widmayer, F. Triguero, R. Morales, M. Hennessy, S. Eidenbenz, and R. Conejo (Eds.), *Proceedings of the 29th International Colloquium on Automata, Languages, and Programming (ICALP)*, Málaga, Spain, Volume 2380 of *Lecture Notes in Computer Science*, pp. 123–134. Springer, Heidelberg.

Fotakis, D., S. Kontogiannis, and P. Spirakis (2004). Selfish unsplittable flows. In *Proceedings of the 31th International Colloquium on Automata, Languages, and Programming (ICALP)*, Turku, Finland, Lecture Notes in Computer Science. Springer, Heidelberg. To appear.

Frank, M. (1981). The Braess paradox. *Mathematical Programming 20*, 283–302.

Frank, M. and P. Wolfe (1956). An algorithm for quadratic programming. *Naval Research Logistics Quarterly 3*, 95–110.

Friedman, E. J. (2003). Genericity and congestion control in selfish routing. Working Paper. Available at `http://www.orie.cornell.edu/~friedman/pfiles/genroute.pdf`.

Friesz, T., D. Bernstein, N. Mehta, R. Tobin, and S. Ganjalizadeh (1994). Day-to-day dynamic network disequilibria and idealized traveler information systems. *Operations Research 42*, 1120–1136.

Friesz, T. L. (1985). Transportation network equilibrium, design and aggregation: Key development and research opportunities. *Transportation Research, Part A 19A*, 413–427.

Friesz, T. L., D. Bernstein, T. E. Smith, R. L. Tobin, and B.-W. Wie (1993). A variational inequality formulation of the dynamic network user equilibrium problem. *Operations Research 41*, 179–191. Special Issue on Stochastic and Dynamic Models in Transportation.

Fudenberg, D. and J. Tirole (1991). *Game Theory.* MIT Press, Cambridge, MA.

Gairing, M., T. Lücking, M. Mavronicolas, B. Monien, and P. Spirakis (2003). Extreme Nash equilibria. In C. Blundo and C. Laneve (Eds.), *Proceedings of the 8th Italian Conference on Theoretical Computer Science (ICTCS)*, Bertinoro, Italy, Volume 2841 of *Lecture Notes in Computer Science*, pp. 1–20. Springer, Heidelberg.

Gairing, M., T. Lücking, M. Mavronicolas, B. Monien, and P. Spirakis (2004). The structure and complexity of extreme Nash equilibria. *Theoretical Computer Science*. To appear.

Garey, M. R. and D. S. Johnson (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA.

Gartner, N. H., S. B. Gershwin, J. D. C. Little, and P. Ross (1980). Pilot study of computer-based urban traffic management. *Transportation Research, Part B 14B*, 203–217.

Gibert, A. (1968). A method for the traffic assignment problem. Technical Report LBS-TNT-95, Transportation Network Theory Unit, London Business School, London.

Grötschel, M., L. Lovász, and A. Schrijver (1993). *Geometric Algorithms and Combinatorial Optimization*. Springer, Berlin.

Hagstrom, J. N. and R. A. Abrams (2001). Characterizing Braess's paradox for traffic networks. In *Proceedings of IEEE Conference on Intelligent Transportation Systems*, Oakland, CA, pp. 836–842. IEEE Computer Society Press, Los Alamitos, CA.

Hall, R. W. (1996). Route choice and advanced traveller information systems on a capacitated and dynamic network. *Transportation Research, Part C 4C*, 289–306.

Haurie, A. and P. Marcotte (1985). On the relationship between Nash-Cournot and Wardrop equilibria. *Networks 15*, 295–308.

Hearn, D. W. (1980). Bounding flows in traffic assignment models. Technical Report 80-4, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL.

Hearn, D. W. and M. V. Ramana (1998). Solving congestion toll pricing models. In P. Marcotte and S. Nguyen (Eds.), *Equilibrium and Advanced Transportation Modeling*, pp. 109–124. Kluwer Academic, Boston, MA.

Hearn, D. W. and J. Ribera (1980). Bounded flow equilibrium problems by penalty methods. In *Proceedings of the IEEE International Conference on Circuits and Computers*, Volume 1, pp. 162–166. Institute of Electrical and Electronics Engineers, New York, NY.

Hearn, D. W. and J. Ribera (1981). Convergence of the Frank-Wolfe method for certain bounded variable traffic assignment problems. *Transportation Research, Part B 15B*, 437–442.

Henry, J. J., C. Charbonnier, and J. L. Farges (1991). Route guidance, Individual. In M. Papageorgiou (Ed.), *Concise Encyclopedia of Traffic & Transportation Systems*, pp. 417–422. Pergamon Press, Oxford.

Holmberg, K. and D. Yuan (2003). A multicommodity network-flow problem with side constraints on paths solved by column generation. *INFORMS Journal on Computing 15*, 42–57.

Hoppe, B. and É. Tardos (1994). Polynomial time algorithms for some evacuation problems. In *Proceedings of the 5th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Arlington, VA, pp. 433–441. SIAM, Philadelphia, PA.

Hoppe, B. and É. Tardos (2000). The quickest transshipment problem. *Mathematics of Operations Research 25*, 36–62.

Inouye, H. (1987). Traffic equilibria and its solution in congested road networks. In R. Genser (Ed.), *Control in Transportation Sysyems*, Proceedings of the 5th IFAC/IFIP/IFORS Conference, Vienna, Austria, July 1986, pp. 267–272. Pergamon Press, Oxford.

Jahn, O., R. H. Möhring, A. S. Schulz, and N. E. Stier-Moses (2002). System-optimal routing of traffic flows with user constraints in networks with congestion. Technical Report 744-2002, Institut für Mathematik, Technische Universität Berlin, Germany.

Jahn, O., R. H. Möhring, A. S. Schulz, and N. E. Stier-Moses (2004). System-optimal routing of traffic flows with user constraints in networks with congestion. *Operations Research*. To Appear. A preliminary version appeared as Working Paper No. 4394-02, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, 2002.

Jarvis, J. J. and H. D. Ratliff (1982). Some equivalent objectives for dynamic network flow problems. *Management Science 28*, 106–109.

Johari, R. (2004). *Efficiency Loss in Market Mechanisms for Resource Allocation*. Ph. D. thesis, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA.

Johari, R. and J. N. Tsitsiklis (2004). Network resource allocation and a congestion game. *Mathematics of Operations Research*. To appear.

Jorgensen, N. O. (1963). Some aspects of the urban traffic assignment problem. Master's thesis, Institute of Transportation and Traffic engineering, University of California, Berkeley, CA.

Kameda, H., E. Altman, T. Kozawa, and Y. Hosokawa (2001). Braess-like paradoxes in distributed computer systems. *IEEE Transactions on Automatic Control 45*, 1687–1691.

Karakostas, G. and S. Kolliopoulos (2003). Selfish routing in the presence of side constraints. Technical Report CAS-03-13-GK, Dept. of Computing & Software, McMaster University, Hamilton, Ontario, Canada.

Kaufman, D. E., R. L. Smith, and K. E. Wunderlich (1991). An iterative routing/assignment method for anticipatory real-time route guidance. In *Proceedings of the IEEE Vehicle Navigation & Information Systems Conference*, Dearborn, MI, Volume 2, pp. 693–700. Society of Automotive Engineers, Warrendale, PA.

Kaysi, I. A. (1992). *Framework and models for the provision of real-time driver information*. Ph. D. thesis, Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Kaysi, I. A., M. E. Ben-Akiva, and A. de Palma (1995). Design aspects of advanced traveler information systems. In N. H. Gartner and G. Improta (Eds.), *Urban Traffic Networks. Dynamic Flow Modelling and Control*, pp. 59–81. Springer, Berlin.

Kaysi, I. A., M. E. Ben-Akiva, and H. Koutsopoulos (1993). Integrated approach to vehicle routing and congestion prediction for real-time driver guidance. *Transportation Research Record 1408*, 66–74.

Kelly, F. P. (1997). Charging and rate control for elastic traffic. *European Transactions on Telecommunications 8*, 33–37.

Knight, F. H. (1924). Some fallacies in the interpretation of social cost. *Quarterly Journal of Economics 38*, 582–606.

Kohl, J. G. (1841). Der verkehr und die ansiedelungen der menschen in ihrer abhängigkeit von der gestaltung der erdoberfläche. Arnold, Dresden/Leipzig.

Köhler, E., K. Langkau, and M. Skutella (2002). Time-expanded graphs with flow-dependent transit times. In R. H. Möhring and R. Raman (Eds.), *Proceedings of the 10th Annual European Symposium on Algorithms (ESA)*, Rome, Italy, Volume 2461 of *Lecture Notes in Computer Science*, pp. 599–611. Springer, Heidelberg.

Köhler, E. and M. Skutella (2002). Flows over time with load-dependent transit times. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, San Francisco, CA, pp. 174–183. SIAM, Philadelphia, PA.

Korilis, Y. and A. Lazar (1995). On the existence of equilibria in noncooperative optimal flow control. *Journal of the ACM 42*, 584–613.

Koutsoupias, E., M. Mavronicolas, and P. Spirakis (2003). Approximate equilibria and ball fusion. *Theory of Computing Systems 36*, 683–693.

Koutsoupias, E. and C. H. Papadimitriou (1999). Worst-case equilibria. In C. Meinel and S. Tison (Eds.), *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, Trier, Germany, Volume 1563 of *Lecture Notes in Computer Science*, pp. 404–413. Springer, Heidelberg.

Lafortune, S., R. Sengupta, D. E. Kaufman, and R. L. Smith (1991). A dynamical system model for traffic assignment in networks. In *Proceedings of the IEEE Vehicle Navigation & Information Systems Conference*, Dearborn, MI, Volume 2, pp. 701–708. Society of Automotive Engineers, Warrendale, PA.

Larsson, T. and M. Patriksson (1994). Equilibrium characterizations of solutions to side constrained asymmetric traffic assignment models. *Le Matematiche 49*, 249–280.

Larsson, T. and M. Patriksson (1995). An augmented Lagrangean dual algorithm for link capacity side constrained traffic assignment problems. *Transportation Research, Part B 29B*, 433–455.

Larsson, T. and M. Patriksson (1999). Side constrained traffic equilibrium models—analysis, computation and applications. *Transportation Research, Part B 33B*, 233–264.

LeBlanc, L. J., R. V. Helgason, and D. E. Boyce (1985). Improved efficiency of the Frank-Wolfe algorithm for convex network programs. *Transportation Science 19*, 445–462.

Leventhal, T., G. Nemhauser, and L. Trotter (1973). A column generation algorithm for optimal traffic assignment. *Transportation Science 7*, 168–176.

Lin, H., T. Roughgarden, and É. Tardos (2004). On braess's paradox. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, New Orleans, LA, pp. 333–334. SIAM, Philadelphia, PA.

Lücking, T., M. Mavronicolas, B. Monien, M. Rode, P. Spirakis, and I. Vrto (2003). Which is the worst-case Nash equilibrium? In B. Rovan and P. Vojtáš (Eds.), *Proceedings of the 28th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, Bratislava, Slovakia, Volume 2747 of *Lecture Notes in Computer Science*, pp. 551–561. Springer, Heidelberg.

Magnanti, T. L. (1984). Models and algorithms for predicting urban traffic equilibria. In M. Florian (Ed.), *Transportation Planning Models*, Proceedings of the course given at the International Center for Transportation Studies (ICTS), Amalfi, Italy, 1982, pp. 153–185. North Holland, Amsterdam.

Mahmassani, H. S., T.-Y. Hu, S. Peeta, and A. Ziliaskopoulos (1994). Development and testing of dynamic traffic assignment and simulation procedures for ATIS/ATMS applications. Technical Report DTFH61-90-R-0074-FG, Center for Transportation Research, University of Texas at Austin.

Mahmassani, H. S. and S. Peeta (1993). Network performance under system optimal and user equilibrium dynamic assignments: Implications for Advanced Traveler Information Systems. *Transportation Research Record 1408*, 83–93.

Mahmassani, H. S. and S. Peeta (1995). System optimal dynamic assignment for electronic route guidance in a congested traffic network. In N. H. Gartner and G. Improta (Eds.), *Urban Traffic Networks. Dynamic Flow Modelling and Control*, pp. 3–37. Springer, Berlin.

Marcotte, P., S. Nguyen, and A. Schoeb (2004). A strategic flow model of traffic assignment in static capacitated networks. *Operations Research 52*, 191–212. To appear. A preliminary version appeared as Technical Report 2000-10, Centre de Recherche sur les Transports.

Maugeri, A. (1994). Optimization problems with side constraints and generalized equilibrium principles. *Le Matematiche 49*, 305–312.

Mavronicolas, M. and P. Spirakis (2001). The price of selfish routing. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, Hersonissos, Greece, pp. 510–519. ACM Press, New York, NY.

Mehlhorn, K. and M. Ziegelmann (2000). Resource constrained shortest paths. In M. Paterson (Ed.), *Proceedings of the 8th Annual European Symposium on Algorithms (ESA)*, Saarbrücken, Germany, Volume 1879 of *Lecture Notes in Computer Science*, pp. 326–337. Springer, Heidelberg.

Merchant, D. K. and G. L. Nemhauser (1978). A model and an algorithm for the dynamic traffic assignment problems. *Transportation Science 12*, 183–199.

Nagurney, A. (1993). *Network Economics: A Variational Inequality Approach.* Kluwer Academic, Dordrecht, The Netherlands.

Nash, J. F. (1951). Noncooperative games. *Annals of Mathematics 54*, 128–140.

Nesterov, Y. (2000). Stable traffic equilibria: Properties and applications. *Optimization and Engineering 3*, 29–50.

Nesterov, Y. and A. de Palma (2000). Stable dynamics in transportation systems. CORE Discussion Paper 00/27, Université Catholique de Louvain, Louvain-la-Neuve, Belgium.

Orda, A., R. Rom, and N. Shimkin (1993). Competitive routing in multi-user communication networks. *IEEE/ACM Trans. Networking 1*, 614–627.

Osborne, M. J. and A. Rubinstein (1994). *A Course in Game Theory.* MIT Press, Cambridge, MA.

Owen, G. (1995). *Game Theory.* Academic Press, San Diego.

Papadimitriou, C. H. (2001). Algorithms, games, and the Internet. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, Hersonissos, Greece, pp. 749–753. ACM Press, New York, NY.

Papadimitriou, C. H. and M. Yannakakis (1994). On complexity as bounded rationality. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing (STOC)*, Montreal, Canada, pp. 726–733. ACM Press, New York, NY.

Papageorgiou, M. (1990). Dynamic modeling, assignment, and route guidance in traffic networks. *Transportation Research, Part B 24B*, 471–496.

Patriksson, M. (1994). *The Traffic Assignment Problem: Models and Methods.* VSP, Utrecht, The Netherlands.

Payne, H. J. and W. A. Thompson (1975). Traffic assignment on transportation networks with capacity constraints and queuing. Presented at the 47th ORSA/TIMS meeting, Chicago, IL.

Peeta, S. (1994). *System optimal dynamic traffic assignment in congested networks with advanced information systems.* Ph. D. thesis, University of Texas at Austin, Austin, TX.

Peeta, S. and A. Ziliaskopoulos (2001). Foundations of dynamic traffic assignment: The past, the present and the future. *Networks and Spatial Economics 1*, 233–265.

Perakis, G. (2004). The "price of anarchy" under nonlinear and asymmetric costs. In G. Nemhauser and D. Bienstock (Eds.), *Proceedings of the 10th Conference on Integer Programming and Combinatorial Optimization (IPCO)*, New York, NY, Volume 3064 of *Lecture Notes in Computer Science.* Springer, Heidelberg. To Appear.

Pigou, A. C. (1920). *The Economics of Welfare.* Macmillan, London.

Powell, W. B., P. Jaillet, and A. Odoni (1995). Stochastic and dynamic networks and routing. In M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser (Eds.), *Networks*, Volume 4 of *Handbook in Operations Research and Management Science*, pp. 141–295. Elsevier Science, Amsterdam.

Prager, W. (1954). Problems of traffic and transportation. In *Proceedings of the Symposium on Operations Research in Business and Industry*, pp. 105–113. Midwest Research Institute, Kansas City, MO.

Qiu, L., Y. R. Yang, Y. Zhang, and S. Shenker (2003). On selfish routing in Internet-like environments. In *Proceedings of the 2003 Conference on Applications, Technologies,* *Architectures, and Protocols for Computer Communications (SIGCOMM)*, Karlsruhe, Germany, pp. 151–162. ACM Press, New York, NY.

Ribeiro, C. and M. Minoux (1986). Solving hard constrained shortest path problems by Lagrangean relaxation and branch-and-bound algorithms. *Methods of Operations Research 53*, 303–316.

Rosenthal, R. W. (1973). A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory 2*, 65–67.

Roughgarden, T. (2002). How unfair is optimal routing? In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, San Francisco, CA, pp. 203–204. SIAM, Philadelphia, PA.

Roughgarden, T. (2003a). Personal communication.

Roughgarden, T. (2003b). The price of anarchy is independent of the network topology. *Journal of Computer and System Sciences 67*, 341–364.

Roughgarden, T. (2004a). Designing networks for selfish users is hard. *Journal of Computer and System Sciences*. To appear. A preliminary version appeared in FOCS 2001.

Roughgarden, T. (2004b). The maximum latency of selfish routing. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, New Orleans, LA, pp. 973–974. SIAM, Philadelphia, PA.

Roughgarden, T. (2004c). Stackelberg scheduling strategies. *SIAM Journal on Computing 33*, 332–350.

Roughgarden, T. and É. Tardos (2002). How bad is selfish routing? *Journal of the ACM 49*, 236–259.

Roughgarden, T. and É. Tardos (2004). Bounding the inefficiency of equilibria in nonatomic congestion games. *Games and Economic Behavior 47*, 389–403.

Schulz, A. S. and N. E. Stier-Moses (2003). On the performance of user equilibria in traffic networks. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Baltimore, MD, pp. 86–87. SIAM, Philadelphia, PA.

Schulz, A. S. and N. E. Stier-Moses (2004). The price of anarchy with elastic demands. In preparation.

Sheffi, Y. (1985). *Urban Transportation Networks.* Prentice-Hall, Englewood, NJ.

Shenker, S. J. (1995). Making greed work in networks: a game-theorectic analysis of switch service disciplines. *IEEE/ACM Transactions on Networking 3*, 819–831.

Smith, M. J. (1979). The existence, uniqueness and stability of traffic equilibria. *Transportation Research, Part B 13B*, 295–304.

Steenbrink, P. A. (1974). *Optimization of Transport Networks.* John Wiley & Sons, London.

Steinberg, R. and W. I. Zangwill (1983). The prevalence of Braess' paradox. *Transportation Science 17*, 301–318.

Texas Transportation Institute (2002). Urban mobility study. Available at `http://mobility.tamu.edu/ums`.

Transport for London (2004). Congestion pricing: 6 months on. Available at http://www.tfl.gov.uk/.

Vavasis, S. A. (1991). *Nonlinear Optimization: Complexity Issues.* Oxford University Press, New York, NY.

Vetta, A. (2002). Nash equilibria in competitive societies with applications to facility location, traffic routing and auctions. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, Vancouver, Canada, pp. 416–428. IEEE Computer Society Press, Los Alamitos, CA.

Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. *Proceedings of the Institution of Civil Engineers, Part II, Vol. 1*, 325–378.

Weitz, D. (2001). The price of anarchy. Unpublished manuscript. Available at http://www.cs.berkeley.edu/~dror/games_proj.ps.

Wie, B.-W., R. L. Tobin, D. Bernstein, and T. L. Friesz (1995). A comparison of system optimum and user equilibrium dynamic traffic assignments with schedule delays. *Transportation Research, Part C 3C*, 389–411.

Yang, H. and S. Yagar (1994). Traffic assignment and traffic control in general freeway-arterial corridor systems. *Transportation Research, Part B 28B*, 463–486.

Yang, T. A., S. Shekhar, B. Hamidzadeh, and P. A. Hancock (1991). Path planning and evaluation in IVHS databases. In *Proceedings of the IEEE Vehicle Navigation & Information Systems Conference*, Dearborn, MI, Volume 1, pp. 283–290. Society of Automotive Engineers, Warrendale, PA.

Index