Multicommodity Routing Problems

Selfish Behavior and Online Aspects

vorgelegt von
Dipl.-Math. Tobias Harks
aus Dorsten

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

Doktor der Naturwissenschaften
– Dr. rer. nat. –

genehmigte Dissertation

Vorsitzender:	Prof. Dr. Fredi Tröltzsch
Berichter:	Prof. Dr. Martin Grötschel
	Prof. Dr. Rolf H. Möhring
	Prof. Dr.-Ing. Adam Wolisz

Tag der wissenschaftlichen Aussprache: 9. Juli 2007

Berlin 2007
D 83

Acknowledgments

I am very grateful to all my friends and colleagues at ZIB, who supported me during the time of writing this thesis. Special thanks go to my *Doktorvater* and supervisor Prof. Dr. Martin Grötschel for making this work possible by offering me a position and giving me the freedom to work on topics that I have been interested in. The financial support for this work was granted through a fellowship from the *Deutsche Forschungsgemeinschaft* within the graduate school *MAGSI*.

I am indebted to Marc E. Pfetsch and Stefan Heinz for their collaboration and precise proofreading. Some results of Chapter 3 and 5 are joint work and an outcome of our fruitful discussions. Their doors and minds were always open for me. I am also grateful to Hans-Florian Geerdes, Tjark Vredeveld, Thomas Schlechte, Christian Raak, Philipp Friese, Uli Menne, Steve Ward, and Tobias Poschwatta for their support, collaboration, and proofreading. Thanks also go to Ariffin Yahaya for bringing the basic problem of Chapter 3 to my attention. Our very productive meetings resulted in a joint work that forms part of Chapter 3. I am indebted to László A. Végh, who contributed to parts of Chapter 5. I also thank Prof. Dr. Rolf Möhring and Prof. Dr.-Ing. Adam Wolisz for their willingness to take the assessments. Their suggestions and ideas during our meetings within the graduate school MAGSI were very valuable for me. I also would like to thank Prof. Dr. Helmut Maurer for introducing me to the subtleties of convex optimiztion and optimal control.

Thanks go also to a very nice group of people at the Democritos University in Xanthi, Greece. During my stay there, I experienced beside excellent collaborations a very warm hospitality. I am indebted to Aggeliki Tsioliaridou, Lefteris Mamatas, Ioannis Psaras, Christos Samaras, Panagiotis Papadimitriou, Napoleon, and Prof. Dr. Vassilis Tsaoussidis.

I would like to express my gratitude to friends, who contributed to this work, though not directly, yet more than they might think: Roman Böckmann, Sebastian Vehlken, Rüdiger Wüllner, Hans Bohnet, and Jan Trowitzsch. I am very grateful to my family for their constant support: Jürgen, Markus, and my parents Elisabeth and Willi. I am deeply indebted to Ines Jungebloed for her constant and unshakeable backup.

Berlin, April 2007 Tobias Harks

Contents

Chapter 1

Introduction

A standard problem in network optimization is to find a routing of traffic demands from sources to destinations using a given network infrastructure. This problem is also known as a *multicommodity flow problem* or *traffic assignment problem*. A cost is associated with every arc of the network, which is usually a function of the arc flow. Most of the optimization models assume a central planer controlling the whole system and determining the best possible routing accordingly. This routing is also called the *system optimum*. For many real-world applications, this assumption is problematic with respect to several aspects: It is not always assured that (i) a central planer has access to the necessary information (*information problem*), (ii) the best possible solution is efficiently computable even if the needed information is accessible (*complexity problem*), (iii) the individual traffic sources agree to a proposed solution (*implementation problem*).

A tremendous amount of effort has been invested in designing efficient routing algorithms to cope with the above problems. As an example for the information problem, consider a traffic assignment problem, where demands arrive sequentially in time in an online fashion. An algorithm that routes these demands without knowledge about future demands is called an *online algorithm*. The common theoretical concept to evaluate the efficiency of an online algorithm is based on *competitive analysis* coming from the online optimization field. An online algorithm is called competitive, if its cost is never larger than a constant factor times the cost of an optimal offline solution. Another research area that covers the complexity problem is concerned with deriving efficient algorithms for solving (NP-hard) optimization problems. Of particular interest is the notion of an *approximation ratio* for a heuristic to solve such optimization problems. The approximation ratio is defined as the largest ratio of the objective value obtained by the solution of the heuristic and that of an optimal solution. The implementation problem can be analyzed within the algorithmic game theory field. Here, one tries to quantify the efficiency loss caused by selfish users compared to the system optimum. The cost of this lack of coordination has been coined "price of anarchy" by Koutsoupias and Papadimitriou in [62]. While the approximation ratio and competitive ratio measure the worst case loss in solution quality due to insufficient computing power and information, respectively, the price of anarchy measures the worst case loss due to insufficient ability to control and coordinate the actions of selfish individuals.

All three issues are exemplified by several practical applications that have motivated the topics covered in this thesis. For instance, billions of packets traverse the world wide web along routes that are decided on by Internet routing protocols. This routing is done in an online fashion without knowledge about future traffic changes. The size of the Internet and the heterogeneity of Internet applications contribute to the computational complexity of finding the best possible routing. Furthermore, a centrally coordinated implementation usually contradicts security requirements of Internet users. Another example for the implementation problem is the road traffic network, where the majority of traffic follows routes that are chosen based on selfish interests of the individuals. It is well known that some users would have to take long detours in a system optimal routing, which makes such a solution unattractive for the affected users.

The main topic of this thesis is to study multicommodity flow problems that exhibit a combination of the afore mentioned three problems. In particular, we focus on online multicommodity routing problems, selfish routing problems, and a combination of these two problems. Thus, the theoretical concepts that we use to analyze the corresponding routing patterns stems from competitive analysis and bounding the price of anarchy. Indirectly, these concepts also provide an approximation ratio, since a solution produced by an online algorithm or a solution produced by selfish individuals constitutes an approximation for the optimal solution of an optimization problem.

1.1 Online Multicommodity Flow Problems

In the first part of this thesis, we study online multicommodity routing problems, where demands have to be routed sequentially in a network. The cost of a flow is determined by dynamic load dependent price functions on links. We make four crucial assumptions: (i) demands for commodities are revealed in an *online* fashion and have to be routed immediately; (ii) demands can be split along several paths; (iii) once a demand is routed, no rerouting is allowed; (iv) the routing cost on an arc is given by the integral over the arc flow with respect to the corresponding price function. Since at the time of routing a commodity, future demands are not known, this yields an online optimization problem that we call the *Online Multicommodity Routing Problem*.

This problem arises in an inter-domain resource market in which multiple service providers offer network resources (capacity) to enable Internet traffic with specific Quality of Service (QoS) constraints, see for example Yahaya and Suda [88] and Yahaya, Harks, and Suda [90]. In such a market, each service provider advertises prices for resources that he wants to sell. We assume that prices are determined by load dependent price functions. Buying providers reserve capacity along paths to route demand (coming from own customers) from sources to destinations via domains of other providers. The routing of a demand along paths is fixed by establishing a contract between the source domain and all domains along the paths. Prices in the market, however, are only valid for a predefined bundle size, that is, after the routing of flow with this bundle size, the arc prices are updated. In the limiting case, where the bundle size tends to zero, the routing cost on an arc is given by the integral over the arc flow with respect to the corresponding price function.

Contributions (Chapter 3)

We investigate such multicommodity online routing problems and allow for arbitrary continuous and nondecreasing load dependent price functions defining the routing costs. We investigate a greedy online algorithm, called SEQ, for this setting and investigate, in which cases this algorithm is competitive. Our main finding is that for polynomial price functions with nonnegative coefficients, the competitive ratio of SEQ can be bounded by a constant factor that only depends on the maximum degree of the polynomials but is independent of the network topology and demand sequence. For the single-source single-destination case, we show that this algorithm is optimal. Without restrictions on the price functions and network, no algorithm is competitive. We also investigate a variant in which the demands have to be routed unsplittably. In this case, the offline problem is NP-hard. As in the splittable case, in general there exists no competitive deterministic online algorithm. For linear price functions, any deterministic online algorithm has a competitive ratio of at least 2. Finally, we present a computational study for unsplittable routings for a realistic network topology and stochastically generated demands. Our empirical findings state that the efficiency loss is significantly smaller in this case compared to the provable upper bounds for the splittable online routing SEQ. The online algorithm SEQ and the ONLINEMCRP can be viewed as a first step towards a methodology for analyzing the efficiency of general inter-domain routing strategies. These results are presented in Chapter 3.

1.2 Network Games

Second, we study the impact of selfish behavior on social welfare in network games. We are interested in the degradation of system performance if players select routes based on selfish interest. Consider a network of arcs that are used by individuals to route demand from sources to destinations. A common approach is to model congestion on arcs by nondecreasing latency functions mapping the flow on an arc to the time needed to traverse this arc. Since individuals share the same network, congestion effects on arcs generate interdependencies between the routing decisions. In this regard, non-cooperative game theory provides the appropriate concepts to analyze such interdependencies. In a non-cooperative game, players compete for shared resources and the utility of each individual player depends on the number of players that choose the same or some overlapping strategy, see Rosen [78]. In the network routing context, the strategies correspond to the available routes and the utility of a player is its total travel time. A classical approach to describe the outcome of a non-cooperative game is to analyze an equilibrium situation. The most popular notion of such an equilibrium is the *Nash equilibrium*: a stable point from which no individual has an incentive to deviate unilaterally. In nonatomic network games in which a single individual player has only a negligible impact on the travel time of others, Wardrop [87] characterized such an equilibrium in his first principle as follows. All path flows between a single source and a single destination have equal latency. A Wardrop equilibrium can be interpreted as a Nash equilibrium in this case.

A fundamental question that has already been raised in 1920 by Pigou [75] and later on in the 1950's by Wardrop [87] and Beckmann, McGuire and Winsten [11] is the following: How efficient is the performance of a Nash equilibrium compared to the best possible outcome? As already noted, the cost of this lack of coordination is called *price of anarchy*. For the Wardrop traffic model, Roughgarden and Tardos [84] proved in a seminal paper that the total travel time of a flow at Nash equilibrium does not deviate too much from the minimum total travel time. In particular, they proved that the price of anarchy is bounded by 4/3 provided affine linear latency functions are considered. By introducing the so called *anarchy value* $\alpha(\mathcal{L})$ for a class $\mathcal{L}$ of latency functions, Roughgarden [81] proved the first tight bounds on the price of anarchy for general polynomial latency functions. Correa, Schulz, and Stier-Moses [24] introduced a different parameter $\beta(\mathcal{L})$ that allows to relax some previous assumptions on allowable latency functions. They proved that their bound implies all bounds of Roughgarden by using the relation $\alpha(\mathcal{L}) = \big(1 - \beta(\mathcal{L})\big)^{-1}$.

Even though we have just argued that the outcome of a Nash equilibrium is not too inefficient, there has been a recent trend towards using route guidance devices for improving the individual travel time. The current position of each driver is determined via the *Global Positioning System* (GPS) at the beginning of a trip. A central computer calculates then an "optimal" route for this trip based on digital maps and on available knowledge of traffic congestion on the streets. In game theoretic language, a route guidance operator is an *atomic player* since a significant (non negligible) part of the entire demand is controlled. Roughgarden [83] and Correa, Schulz, and Stier-Moses [25] claimed that the price of anarchy in an atomic network game does not exceed that of the corresponding nonatomic one. Interestingly, this turned out to be wrong, as reported by Cominetti, Correa, and Stier-Moses in [23]. Based on the work of Catoni and Pallotino [19], they presented an example in which the price of anarchy in a network game with atomic players is larger than that of the corresponding nonatomic game. Moreover, they showed that the cost for an atomic player may even increase compared to the nonatomic game. Such a counter-intuitive phenomenon can also arise from the perspective of single individuals: a nonatomic player competing with an atomic player may face lower cost compared to the situation in which the atomic player is replaced by nonatomic ones. Cominetti, Correa, and Stier-Moses showed that the price of anarchy for the atomic network game can be bounded for special latency functions. In particular, they proved upper bounds of $1.5, 2.56$, and 7.83 on the price of anarchy for affine linear, squared, and cubic latency functions with nonnegative coefficients, respectively. For polynomials with nonnegative coefficients and higher degree, their approach fails to generate upper bounds on the price of anarchy.

Contributions (Chapter 4)

For network games with nonatomic players, we introduce the value $\omega(\mathcal{L}, \lambda)$ for bounding the price of anarchy. This value generalizes the anarchy value $\alpha(\mathcal{L})$ and the value $\beta(\mathcal{L})$. Using our value, we reprove the existing tight bounds on the price of anarchy and present a novel proof for monomial latency functions showing that the price of anarchy is one in this case.

For network games with atomic players, we improve all previously known bounds for polynomial latency functions with nonnegative coefficients, except for affine linear latency functions. These results are presented in Chapter 4.

1.3 Online Network Games

Combining the online aspect with selfish behavior of individuals, we investigate an online routing problem called *online network games*. In this problem, we assume a sequence of network games $\sigma = (1, \ldots, n)$ that are released consecutively in time in an online fashion. A network game is characterized by a set of commodities that have to be routed in a given network. Arcs in the network are equipped with load dependent latency functions defining the routing cost. By the time of routing commodities of game i, future games $i + 1, \ldots, n$ are not known. We further assume that once commodities of a game are routed, this routing remains fixed, that is, the routings are irrevocable. We analyze two online algorithms, called NSeqNash and ASeqNash. These algorithms produce a flow consisting of a sequence of Nash equilibria for the corresponding games with nonatomic and atomic players, respectively. As usual, we analyze the efficiency of an online algorithm in terms of competitive analysis.

The online variant of network games is motivated by the application of selfish routing to the source routing concept in telecommunication networks, see Qiu, Yang, Zhang, and Shenker [76] and Friedman [42] for an engineering perspective and Roughgarden [80] and Altman, Basar, Jimenez, and Shimkin [5] for a theoretical perspective on this topic. In the source routing model, sources are responsible for selecting paths to route data to the corresponding destination. The arcs in the network advertise their current status that is based on the current congestion situation. If the costs on arcs correspond to the expected delay, minimum cost routing is a natural goal for real-time applications.

As described in the last section, the main focus of the line of research that studies source routing is to quantify the price of anarchy. Here, one assumption is crucial: if the traffic matrix changes, all sources may possibly change their routes and form a new equilibrium. This assumption, however, has some important implications: Each source would have to *continuously* maintain the current state of all available routes, which in turn introduces additional traffic overhead by continuously signaling this needed information. Furthermore, frequent rerouting attempts during data transmission may not only produce transient load oscillations but may also interfere with the widely used congestion control protocol TCP that controls the data rate, as reported by La, Walrand, and Anantharam in [63]. For these reasons, frequent rerouting attempts in reaction to traffic changes in the network are not necessarily beneficial. Time critical applications, such as Internet telephony or video streaming may suffer severe performance degradation.

Contributions (Chapter 5)

In this regard, we propose and investigate a new model, called *Online Network Games*, in which sources starting at the same time select their routes *only* during connection setup phase. We then study the extreme case in which flows fix their routing decisions once they are at equilibrium. Thus, continuously gathering information about the state of available routes is not necessary after this initial routing game. Relying on competitive analysis, we analyze online algorithms that produce a flow that is at Nash equilibrium for every game out of a sequence of games. The cost function is given by the total travel cost after all games have been played. Our main result states that for polynomial latency functions with nonnegative coefficients, the competitive ratio of both NSeqNash, andASeqNash can be bounded by a constant factor, which depends on the maximum degree. This result holds independently of the network topology or game sequence. We also prove lower bounds. In particular, we show that for a sequence of two network games and affine linear latency functions, our upper bound for the NSeqNash is tight. Furthermore, we prove for a given sequence of games and parallel arcs that the competitive ratio of the online algorithm NSeqNash does not exceed the price of anarchy of a complementary nonatomic network game in which all commodities of the sequence of games are considered at the same time.

1.4 Thesis Organization

After describing the motivation and background for this thesis in Chapter 1, we present in Chapter 2 the basic concept of competitive analysis in online optimization. In Chapter 3, we present the framework OnlineMCRP in which we study the online algorithm Seq. In Chapter 4 we focus on network games with nonatomic and atomic players, respectively. Finally, we combine network games with online aspects in Chapter 5.

We note that the "Contribution and Chapter Outline" section at the beginning of Chapter 3, 4, and 5 gives an overview and road map about the results presented in that chapter. We further recommend that Chapter 4 is read prior to Chapter 5. Except for introducing the notation for multicommodity flow problems, Chapter 4 and 5 can be read independently from Chapter 3.

Chapter 2

Preliminaries

In this chapter, we introduce the theoretical concept for analyzing online algorithms that cope with incomplete information. In the first section, we introduce the concept of online optimization by means of a request answer game as defined by Ben-David et al. in [12]. The second section deals with the concept of competitive analysis that we are going to use throughout the thesis.

2.1 Online Optimization

Static optimization approaches assume complete knowledge about all problem data in advance. These problems are also called *offline optimization problems*. Online optimization problems are a special class of optimization problems in which the input instances are not given completely in advance. Instead, an instance arises step-by-step and decisions have to be made based only on the information revealed so far. Each decision leads to a cost or profit and the task is to minimize the total cost or to maximize the gained profit. In this thesis only minimization problems are considered. Therefore, all definitions in this section refer to minimization problems. However, the definitions can easily be adapted for maximization problems.

Most online optimization problems can be formalized as a request-answer game, which was introduced by Ben-David et al. in [12].

Definition 2.1 (Request-Answer Game)
A *request-answer game* consists of a request set $\mathcal{R}$, a nonempty and finite answer set $\mathcal{A}$, and cost functions $\text{cost}_n : \mathcal{R}^n \times \mathcal{A}^n \to \mathbb{R}^+ \cup \{\infty\}$ for $n \in \mathbb{N}$. Let $\mathcal{C}$ denote the set of functions cost_n for all $n \in \mathbb{N}$. An instance is given by a request sequence $\sigma = r_1, r_2, \ldots, r_n$ of $n \in \mathbb{N}$ requests from $\mathcal{R}$. The task is to find an answer sequence $(a_1, a_2, \ldots, a_n) \in \mathcal{A}^n$ such that the cost $\text{cost}_n(r_1, r_2, \ldots, r_n, a_1, a_2, \ldots, a_n)$ is minimized. A request-answer game is given by the triple $(\mathcal{R}, \mathcal{A}, \mathcal{C})$.

A request-answer game itself does not define an online optimization problem since no restriction is made on the way the answers have to be given. An online algorithm has to serve a request right after it arises according to the specific rules of a request-answer game. We remark here that it is common in the literature to assume a finite answer set $\mathcal{A}^i$. In this thesis, however, the answer sets that we consider are continuous sets containing infinitely many elements. As stated by Borodin and El Yaniv [15], the finiteness requirement is not of conceptual importance for the principles of competitive analysis. They show that an infinite answer set can be approximated by a sufficiently large finite answer set.

Definition 2.2 (Deterministic Online Algorithm)
Let $(\mathcal{R}, \mathcal{A}, \mathcal{C})$ be a request-answer game. A *deterministic online algorithm* ALG is a sequence of functions $f_1, f_2, \ldots,$ where $f_i : \mathcal{R}^i \to \mathcal{A}$. If $\sigma = r_1, r_2, \ldots, r_n$ is a sequence of $n \in \mathbb{N}$ requests from $\mathcal{R}$, then the output of ALG for this sequence is

$$\textsc{Alg}[\sigma] = (a_1, a_2, \ldots, a_n) \in \mathcal{A}^n, \quad \text{where } a_i = f_i(r_1, r_2, \ldots, r_i).$$

The cost incurred by ALG on σ is denoted by $\textsc{Alg}(\sigma)$ and defined as

$$\textsc{Alg}(\sigma) = \text{cost}_n(\sigma, \textsc{Alg}[\sigma]).$$

Note that the answer a_i may only depend on the requests $r_1, r_2, \ldots, r_i$ for $i = 1, 2, \ldots, n$. Therefore, the definition of a deterministic online algorithm meets the requirement that such algorithms have to make decisions based only on partial information.

Besides the class of deterministic online algorithms, there exists the class of *randomized online algorithms*. These algorithms use a probability distribution over a set of deterministic online algorithms to generate an answer for a given request. Therefore, the answer sequence as well as the cost are random variables. Even though we will not use the concept of randomized online algorithms in this thesis, we briefly present the main ideas for completeness.

Definition 2.3 (Randomized Online Algorithm)
A *randomized online algorithm* RALG is a probability distribution over deterministic online algorithms $\textsc{Alg}_x$ (x may be thought of as the coin tosses of the algorithm RALG).

Note that the definition points out that every deterministic online algorithm is a randomized online algorithm with probability 1 on a certain outcome. Hence, the class of deterministic online algorithms is included in the class of randomized online algorithms.

Online algorithms provide for each sequence of requests an answer sequence which comes along with a cost. Usually, the task is to generate an answer sequence that minimizes this cost.

2.2 Competitive Analysis

The standard technique for analyzing the performance of an online algorithm is competitive analysis. This method measures the performance of an online algorithm against an *optimal offline solution*. An optimal offline algorithm has access to the complete input instance in advance and serves it at a minimum cost, called *optimal offline cost*.

Definition 2.4 (Optimal Offline Cost)
Let $(\mathcal{R}, \mathcal{A}, \mathcal{C})$ be a request-answer game and $\sigma = r_1, r_2, \ldots, r_n$ a sequence of $n \in \mathbb{N}$ requests from $\mathcal{R}$. Then the *optimal offline cost* is defined as

$$\textsc{Opt}(\sigma) = \min\{\text{cost}_n(\sigma, a) \mid a \in \mathcal{A}^n\}.$$

2.2.1 Deterministic Online Algorithm

Using competitive analysis the performance of a deterministic online algorithm is measured as follows.

Definition 2.5 (Competitive Deterministic Online Algorithm)
Let $(\mathcal{R}, \mathcal{A}, \mathcal{C})$ be a request-answer game and $c \geq 1$ a real number. A deterministic online algorithm ALG is called *c-competitive* if there exists a number $b \geq 0$ such that

$$\textsc{Alg}(\sigma) \leq c \cdot \textsc{Opt}(\sigma) + b$$

holds for any request sequence σ. If $b = 0$, ALG is called *strictly c-competitive*.

In the reminder of this thesis we omit the term "strictly". For all presented results we have $b = 0$.

Given a deterministic online algorithm ALG, we are interested in the smallest constant $c \geq 1$ such that ALG is $c-$competitive.

Definition 2.6 (Competitive Ratio)
The *competitive ratio* of a deterministic online algorithm ALG is the infimum over all c such that ALG is c-competitive.

Note that the definition does not make any restriction on the computational complexity of a deterministic online algorithm. The only scarcity in competitive analysis comes from lack of information. The concept of competitive analysis is based on a worst case analysis for online algorithms. The performance guarantee must hold for each request sequence. In this regard, competitive analysis can be seen as a game between the online algorithm and a malicious adversary. The malicious adversary tries to generate a request sequence such that the online algorithm performs as "bad" as possible compared to the optimal offline cost. In doing so, the malicious adversary has knowledge about the algorithm. That is, he knows for any request sequence all answers of a deterministic online algorithms in advance.

2.2.2 Randomized Online Algorithm

The answer sequence as well as the cost of a randomized online algorithm are random variables. Therefore, the competitive ratio of a randomized online algorithm depends on the amount of information an adversary has access to. In the standard adversary model, the adversary has knowledge about the probability distribution of a randomized online algorithm but does not know the exact outcome for each request sequence. Hence, an adversary has to choose an entire request sequence before an online algorithm starts processing the chosen sequence. Such an adversary is called *oblivious adversary* in the literature.

Definition 2.7 (Oblivious Adversary)
An *oblivious adversary* has to generate the entire request sequence in advance based only on the description of the randomized online algorithm but before any request is served by the randomized online algorithm.

As mentioned before, the definition of the competitive ratio of a randomized online algorithm depends on the class of allowable adversaries. For the purpose of introducing competitive analysis, we restrict ourselves to an oblivious adversary, which is the weakest of those introduced by Ben-David et al. in [12].

Definition 2.8 (Competitive Randomized Online Algorithm)
Let $(\mathcal{R}, \mathcal{A}, \mathcal{C})$ be a request-answer game and $c \geq 1$ a real number. A randomized online algorithm RALG with a probability distribution X over a set $\{\textsc{Alg}_x\}$ of deterministic online algorithms is said to be *c-competitive* against the oblivious adversary if

$$\mathbb{E}[\textsc{Alg}_x(\sigma)] \leq c \cdot \textsc{Opt}(\sigma)$$

holds for each sequence σ. Here the expression $\mathbb{E}[\textsc{Alg}_x(\sigma)]$ denotes the expectation with respect to the probability distribution X over $\{\textsc{Alg}_x\}$ which defines RALG. The *competitive ratio* of RALG is the infimum over all c such that RALG is c-competitive against the oblivious adversary.

The above definition reduces to Definition 2.5 in the case of a deterministic online algorithm. Since the oblivious adversary is not as powerful as in the deterministic case, randomized online algorithms usually provide a better competitive ratio than deterministic online algorithms.

Of course, lower bounds on the competitive ratio of online algorithms are also of interest. In order to obtain such a lower bound for an online algorithm, a request sequence has to be constructed such that this algorithm performs "bad" compared to the optimal offline cost. Besides a lower bound on the competitive ratio of a certain online algorithm, it is also of interest to find a lower bound which holds for any online algorithm of the considered online optimization problem. In the deterministic case, it is comparatively easy to find suitable request sequences. Since the cost of a randomized online algorithm is a random variable, it can be difficult to bound the competitive ratio from below. In such cases Yao's Principle is an approach to find lower bounds on the competitive ratio of any randomized online algorithms for the considered online problem, see Borodin and El Yaniv [15], Motwani and Raghaven [70], and Albers [4].

Chapter 3

Online Routing Problems

Resource intensive real-time applications, such as video or Internet-telephony, are increasingly dominating traditional Internet traffic, for example e-mail, file transfer, and web-browsing. This causes significant changes regarding the interaction of Internet service providers. Currently, loose *peering agreements* between neighboring domains regulate the data transmission across domain boundaries: Each domain agrees to route messages between any of its two neighboring domains; this routing is done free of charge and on a best effort basis. One of the main reasons why most service providers agree with this policy is because traditional Internet traffic does not require any Quality of Service (QoS). Therefore, the "free through-passing" policy only incurs low additional cost to each service provider.

Future Internet applications, however, pose diverse QoS requirements on the Internet traffic, e.g., bounded packet delay and jitter [1] for video or Internet-telephony, see Gharavi and Partovi [44]. Moreover, users expect that their access provider delivers this type of service. Presently, Internet service providers can offer such services *within* the domains which they control. Still, providing traffic with QoS requirements for other domains is considerably more expensive than the above mentioned traditional Internet applications. As a consequence, service providers are no longer willing to support this service at no cost. Hence, the deployment of end-to-end inter-domain traffic with QoS requires trading and negotiating for resources between different service providers. This opens a new market with a multitude of strategically acting and selfishly behaving participants.

In this regard, a novel inter-domain resource exchange architecture (iREX) for the automated deployment of Internet traffic with QoS requirements has been proposed by Yahaya and Suda [88, 89] and Yahaya, Harks, and Suda [90]. The iREX architecture is based on the "Posted Price Competition" economic model in which providers independently choose prices that are publicly communicated to resource consumers on a take-it-or-leave-it basis, see Abbink [1] for an introduction to this economic model. In the iREX context, domains are

[1] Jitter is defined as the variation of inter packet arrival times.

both resource providers and resource consumers at the same time. Since domains have resources that they can sell, they also need to buy resources to deploy inter-domain traffic with QoS requirements for their own customers. Thus, the iREX architecture establishes a market for Internet traffic with QoS requirements. In this market, each service provider advertises prices for resources that he wants to sell. Buying providers reserve capacity along paths to route demand (coming from own customers) from source to destination via domains of other providers. The routing of a demand along paths is fixed by establishing a contract between the source domain and all domains along the chosen paths. According to Yahaya and Suda [88, 89], we assume that providers determine prices according to predefined load dependent price functions. Such prices, however, are only valid for a predefined demand size (bundle size), that is, routing flow of bundle size prompts an update of arc prices. The reason for this is that scarce resources are priced higher (lower) if the load increases (decreases). In the limiting case, where the bundle size tends to zero, the routing cost on an arc is given by the integral over the arc flow with respect to the corresponding price function. As the user behavior and resulting traffic changes in the Internet are hard to predict, we assume that future demands are unknown.

In this chapter, we model the interaction of service providers via online multicommodity routing problems. In online multicommodity routing problems, commodities of a multicommodity flow have to be routed sequentially in a network. The cost of a flow is determined by dynamic load dependent price functions on links. We assume that the price functions are continuous and nondecreasing. The commodities arrive sequentially in time and have to be routed with lowest cost, since participating providers are assumed to act rational. We make four crucial assumptions for the considered model: (i) demands for commodities are revealed in an *online* fashion; (ii) demands can be split along several paths; (iii) once a demand is routed, no rerouting is allowed; (iv) the routing cost on an arc is given by the integral over the arc flow with respect to the corresponding price function. Since at the time of routing a commodity, future demands are not known, this yields an online optimization problem that we call the *Online Multicommodity Routing Problem* (OnlineMCRP).

We study a greedy online algorithm Seq that routes a newly revealed commodity by solving a convex optimization problem that only depends on the previously routed demands. We investigate cases in which Seq is *competitive*, that is, its cost is at most a constant factor larger than the cost of an optimal offline solution for which all commodities are known. We see Seq and the OnlineMCRP as a first step towards an analytical methodology for analyzing the efficiency of general inter-domain routing strategies.

Multicommodity routing problems have been studied in the context of traffic engineering, see Fortz and Thorup [38, 39]. There, the goal is to route given demands subject to capacity constraints in order to minimize a convex load dependent penalty function. In this setting, a central planer has full knowledge of all demands, which is not the case in our approach.

Another related line of research is the investigation of efficient routing in decentralized noncooperative systems. This has been extensively studied using game theoretic concepts, cf. Roughgarden and Tardos [84], Correa, Schulz, and Stier Moses [24], Altman, Basar, Jimenez, and Shimkin [5], and the references therein. In these works the efficiency of Nash equilibria are studied. Hence, rerouting of demands is allowed in this context. In our model, once a routing decision has been made, it remains unchanged.

The main topic in online routing has been call admission control problems. An overview article about these problems is given by Leonardi in [32]. Perhaps closest is the paper by Awerbuch, Azar, and Plotkin [9], where online routing algorithms are presented to maximize throughput under the assumption that routings are irrevocable. They restrict the analysis to single path routing and present competitive bounds that depend on the number of nodes in the network.

3.1 Contributions and Chapter Outline

We first introduce in Section 3.2 the formal model for the OnlineMCRP which is followed by the definition of the greedy online algorithm Seq and the optimal offline solution.

Then, in Section 3.3 we show using competitive analysis that no online algorithm for the OnlineMCRP is competitive for general networks and price functions. If the price functions and the network are restricted, however, one can obtain competitive results. For affine linear price functions the greedy online algorithm Seq is $\frac{4K^2}{(1+K)^2}$-competitive, where K is the number of commodities, as shown in Section 3.3.1. Furthermore, we prove a lower bound of $\frac{4}{3}$ on the competitive ratio for any deterministic online algorithm in this case. For Seq, we prove a lower bound of $\frac{2K-1}{K}$. For polynomial price functions with nonnegative coefficients, we prove upper and lower bounds on the competitive ratio of Seq that both grow exponentially in the degree of the considered polynomials.

If we restrict the structure of the network to have a single-source and single-destination only, Section 3.4 shows that Seq returns an optimal solution, i.e. Seq is 1-competitive.

We also study the variant of the OnlineMCRP in which the demands have to be routed (unsplittably) along a single path. In Section 3.5, we prove that the corresponding offline problem is NP-hard. We further show that in general no competitive deterministic online algorithm exists. Finally, we present a lower bound of 2 on the competitive ratio for any deterministic online algorithm if the price functions are linear.

These results are preceded by a formal problem description. This includes the optimality conditions for the convex problems that have to be solved by Seq and to determine an optimal offline solution (Section 3.2).

In Section 3.6, we introduce an unsplittable variant of Seq with expiring demands and study its performance via simulating real world networks and traffic demands. This joint work together with Yahaya and Suda [90] provides empirical evidence for the efficiency of the iREX protocol in an inter-domain QoS market. It turns out that the real-world instances perform better than the derived worst case analytical bounds. In other words: Simulations show that for realistic networks and demands, the efficiency of an online single path routing compared to the best possible outcome is significantly smaller than the provable worst case bounds. We close this chapter with further comments and open questions in Section 3.7.

The results for affine linear price functions, single commodity networks, and unsplittable routings are joint work with Heinz and Pfetsch [52]. The computational study is joint work with Yahaya and Suda [90].

3.2 Problem Description

An instance of the *Online Multicommodity Routing Problem* (OnlineMCRP) consists of a directed network $D = (V, A)$ and nondecreasing and continuous price functions $p_a : \mathbb{R}_+ \to \mathbb{R}_+$ for each link $a \in A$. These functions define the price of reserving capacity on a link depending on the current load, as described below. For convenience, we will sometimes use the words arc and link, vertex and node interchangeably. The arcs are ordered pairs of vertices $a = (i, j)$, where the first vertex i represents the tail of an arc and the second vertex j represents the head. Thus, an arc is directed from its tail to its head. We also allow for *parallel* arcs, which means that we allow for several distinct arcs with the same tail and head.

Furthermore, a sequence $\sigma = 1, \ldots, K$ of commodities must be routed one after the other. We assume that $K \geq 2$ and denote the set of commodities by $[K] := \{1, \ldots, K\}$. Each commodity $k \in [K]$ has a demand $d_k > 0$ that has to be routed from a source $s_k \in V$ to a destination $t_k \in V$. We denote vertices $s_1, \cdots, s_K$ as *sources*, that is, these nodes are the source of traffic demand $d_1, \cdots, d_K$. The vertices $t_1, \cdots t_K$ denote destination nodes, where the traffic from the sources terminate.

To shorten notation we use the following convention: When we speak of a sequence $\sigma = 1, \ldots, K$ of commodities, we refer to the full specification $(d_1, s_1, t_1), \ldots, (d_K, s_K, t_K)$.

The routing decision for commodity k is *online*, that is, it only depends on the routings of commodities $1, \ldots, k-1$. Once a commodity has been routed it remains unchanged.

A routing assignment, or *flow*, for commodity $k \in [K]$ is a nonnegative vector $f^k \in \mathbb{R}^A_+$. This flow is *feasible* if for all $v \in V$ holds that

$$\sum_{a \in \delta^+(v)} f^k_a - \sum_{a \in \delta^-(v)} f^k_a = \gamma(v), \tag{3.1}$$

where $\delta^+(v)$ and $\delta^-(v)$ are the arcs leaving and entering v, respectively; furthermore, $\gamma(v) = d_k$ if $v = s_k$, $\gamma(v) = -d_k$ if $v = t_k$, and $\gamma(v) = 0$ otherwise.

Note that splitting of demands is allowed. We say that $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^K)$ is a *multi-commodity flow*.

An alternative formulation uses a *path flow* for each commodity $k \in [K]$. Let $\mathcal{P}_k$ be the set of all paths from s_k to t_k in D. A path flow for commodity k is a nonnegative vector $(f_P^k)_{P \in \mathcal{P}_k}$. The corresponding flow on link $a \in A$ for commodity $k \in [K]$ is then given by

$$f_a^k := \sum_{P \ni a} f_P^k.$$

The aggregated flow of all commodities on link a can be written as

$$f_a := \sum_{k=1}^{K} f_a^k.$$

In the sequel of this thesis, we use the bold notation, i.e. $\boldsymbol{f}^i$, when we refer to a vector of numbers and normal font, i.e. f_a, when referring to a single real number. We define $\mathcal{F}_k$ with $k \in [K]$ to be the set of vectors $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^k)$ such that $\boldsymbol{f}^i$ is a feasible flow for commodities $i = 1, \ldots, k$. If $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^k) \in \mathcal{F}_k$, we say that it is *feasible* for the sequence of commodities $1, \ldots, k$. The entire flow for the sequence $1, \ldots, K$ of commodities is denoted by $\boldsymbol{f} = (\boldsymbol{f}^1, \ldots, \boldsymbol{f}^K)$. Furthermore, the cost of a flow f_a^k on link $a \in A$ of commodity k is defined by

$$C_a^k(f_a^k; f_a^1, \ldots, f_a^{k-1}) := \int_0^{f_a^k} p_a\Big(\sum_{i=1}^{k-1} f_a^i + z\Big)\, dz. \tag{3.2}$$

For convenience, we sometimes write $C_a^k(f_a^k)$ instead of $C_a^k(f_a^k; f_a^1, \ldots, f_a^{k-1})$. Note that $C_a^k(\cdot)$ is a convex function, because $p_a(\cdot)$ is nondecreasing. Furthermore, the flow values $f_a^1, \ldots, f_a^{k-1}$ of previously routed commodities are fixed parameters for the cost function of commodity k.

Remark 3.1. The cost function in (3.2) can be obtained as the limiting case of a single path routing: Assume that demand d_k is split into N equal pieces and the pieces are routed consecutively along a single path. The cost of this path is obtained by evaluating $p_a(\cdot)$ at the flow on link a arising from the previous routings. Let z_a^ℓ be the flow on arc $a \in A$ arising from piece $\ell \in [N]$, i.e., $z_a^\ell = \frac{d_k}{N}$ if a is on the path and $z_a^\ell = 0$ otherwise. Then we have:

$$\lim_{N \to \infty} \sum_{n=1}^{N} p_a\Big(\sum_{i=1}^{k-1} f_a^i + \sum_{\ell=1}^{n-1} z_a^\ell\Big)\, z_a^n = C_a^k(f_a^k; f_a^1, \ldots, f_a^{k-1}),$$

where $f_a^k := \sum_{n=1}^{N} z_a^n$ is a feasible flow for commodity k. Hence, the integral represents the fact that an infinitesimal amount of flow increases the price for each consecutive piece. Figure 3.1 illustrates the arc costs for demand divided into three pieces. Note that the above described single path routing for a discrete bundle size corresponds to the working mechanism of the iREX protocol as specified by Yahaya and Suda [88, 89].

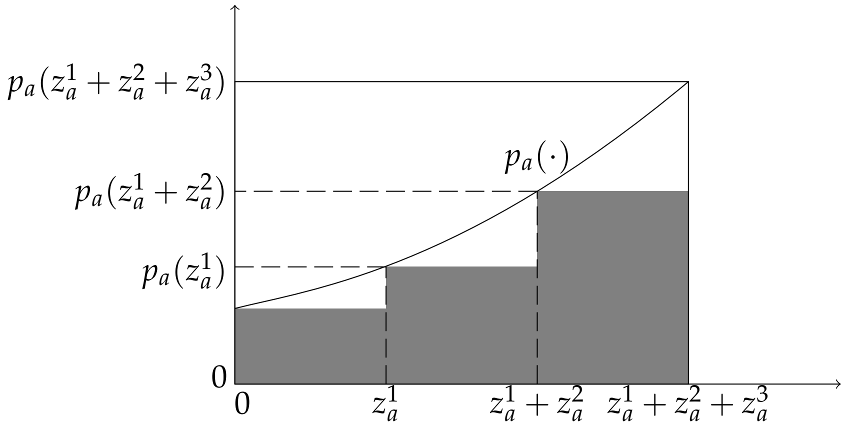

Figure 3.1: Illustration of the motivation for the cost function defined in (3.2). The shaded area corresponds to the arc cost for $N = 3$. For the case $N \to \infty$ the shaded area converges to the exact integral.

Given flows $f^1, \dots, f^{k-1}$, the cost for flow f^k is

$$C^k(f^k; f^1, \dots, f^{k-1}) := \sum_{a \in A} C_a^k(f_a^k; f_a^1, \dots, f_a^{k-1}).$$

To shorten the presentation, we write

$$C^k(f^k) = C^k(f^k; f^1, \dots, f^{k-1}).$$

The total cost over all commodities is defined by

$$C(f) = \sum_{k \in [K]} C^k(f^k).$$

Note that the total cost of all commodities is given by the sum of the individual cost of the single commodities. In this regard, the cost function is separable in the commodities. Loosely speaking, the flow of later commodities do not affect the individual cost of former ones. In the following we derive a nice simplification of the total cost.

Remark 3.2. The total cost can be represented in terms of the aggregated arc flow:

$$C(f) = \sum_{k=1}^{K} C^k(f^k) = \sum_{k=1}^{K} \sum_{a \in A} C_a^k(f_a^k) = \sum_{a \in A} \sum_{k \in [K]} C_a^k(f_a^k) = \sum_{a \in A} \int_0^{f_a} p_a(z)\, dz.$$

The above cost representation implies that the order of commodities plays no role when determining an optimal routing.

3.2.1 The Greedy Online Algorithm Seq

In this section, we study the greedy online algorithm Seq that for a given sequence $\sigma = 1, \dots, K$, sequentially routes the requested demands with minimum cost.

Definition 3.3 (Seq for the OnlineMCRP)
Consider an instance of the OnlineMCRP with a sequence $\sigma = 1, \dots, K$. The deterministic conline algorithm Seq solves for every $k \in [K]$ the following convex program

$$\begin{aligned} \min \quad & C^k(\boldsymbol{f}^k) & & \\ \text{s.t.} \quad & \sum_{a \in \delta^+(v)} f_a^k - \sum_{a \in \delta^-(v)} f_a^k = \gamma(v) & \forall\, v \in V & \qquad (3.3) \\ & f_a^k \geq 0 & \forall\, a \in A, & \end{aligned}$$

where the vectors $\boldsymbol{f}^1, \dots, \boldsymbol{f}^{k-1}$ are fixed by solving the first $k-1$ problems.

Note that the above problem may admit several optimal solutions with the same objective value. Problem (3.3) can be efficiently solved within arbitrary precision in polynomial time (see Grötschel, Lovász, and Schrijver [48]).

Using the relation

$$\frac{\partial C^k}{\partial f_a^k}(\boldsymbol{f}^k) = p_a\Big(\sum_{i=1}^{k} f_a^i\Big),$$

we state in the following lemma necessary and sufficient optimality conditions of the above K problems.

Lemma 3.4. *A feasible flow $\boldsymbol{f} = (\boldsymbol{f}^1, \dots, \boldsymbol{f}^K) \in \mathcal{F}_K$ solves* (3.3) *if and only if for all $k \in [K]$ the following two equivalent conditions are satisfied:*

$$i) \quad \sum_{a \in A} p_a\Big(\sum_{i=1}^{k} f_a^i\Big)\,(f_a^k - x_a^k) \leq 0 \qquad \text{for all feasible flows } \boldsymbol{x}^k \text{ for commodity } k, \qquad (3.4)$$

$$ii) \quad \sum_{a \in P} p_a\Big(\sum_{i=1}^{k} f_a^i\Big) \leq \sum_{a \in Q} p_a\Big(\sum_{i=1}^{k} f_a^i\Big) \qquad \text{for all } P, Q \in \mathcal{P}_k,\ P \text{ flow carrying w.r.t. } \boldsymbol{f}^k. \qquad (3.5)$$

A formal proof of the equivalence of the above statements can be found in Dafermos and Sparrow [27]. In fact, the second condition is simply the Kuhn-Tucker condition of problem (3.3). Then, the only important ingredient is the convexity of the objective function. Since we are using the variational inequality 3.4 throughout the thesis, we only prove that condition (3.4) implies optimality.

Proof. Assume the flow $\boldsymbol{f}^k$ satisfies condition (i). Let $\boldsymbol{x}^k$ be an arbitrary feasible flow. By assumption, the cost function $C^k(\cdot)$ is convex. Hence, we can bound the cost function from below by a linear approximation in $\boldsymbol{f}^k$ (see Figure 3.2 for a graphical illustration of the linear approximation):

$$\sum_{a \in A} C_a^k(x_a^k) \geq \sum_{a \in A} C_a^k(f_a^k) + p_a\Big(\sum_{i=1}^{k} f_a^i\Big)\,(x_a^k - f_a^k).$$

By assumption, the last term is nonnegative, see inequality (3.4). Hence, the cost of the flow $\boldsymbol{x}^k$ is greater than or equal the cost of $\boldsymbol{f}^k$. Since $\boldsymbol{x}^k$ was chosen arbitrarily, the flow $\boldsymbol{f}^k$ solves problem (3.3). □

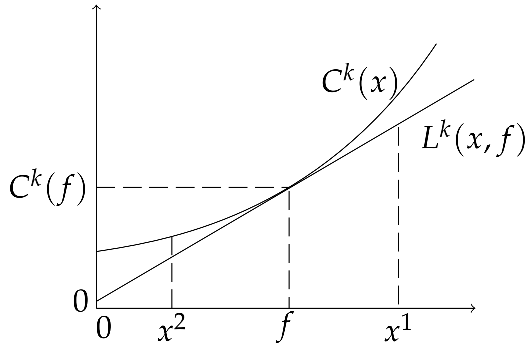

Figure 3.2: Illustration of the linear approximation $L^k(\cdot\,; f)$ in the point f of the convex function $C^k(\cdot)$ with $L^k(x; f) \le C^k(x)$.

3.2.2 The Optimal Offline Solution

An *optimal offline flow* is given by a solution $\boldsymbol{f}^\star$ of the following convex optimization problem:

$$\begin{aligned} \min \quad & C(\boldsymbol{f}) & & \\ \text{s.t.} \quad & \sum_{a\in\delta^+(v)} f_a^k - \sum_{a\in\delta^-(v)} f_a^k = \gamma(v) & & \forall\, v \in V,\ k \in K \\ & f_a^k \ge 0 & & \forall\, a \in A,\ k \in K. \end{aligned} \tag{3.6}$$

We denote by $\textsc{Opt}(\sigma)$ the optimal value of this convex problem.

As Problem (3.3) the above problem can be efficiently solved within arbitrary precision in polynomial time (see Grötschel, Lovász, and Schrijver [48]). Using the relation

$$\frac{\partial C}{\partial f_a^k}(\boldsymbol{f}) = p_a\Big(\sum_{i=1}^{K} f_a^i\Big),$$

the necessary and sufficient optimality conditions of the above problem are the following.

Lemma 3.5. *A flow $\boldsymbol{f} = (\boldsymbol{f}^1, \dots, \boldsymbol{f}^K)$ is offline optimal if and only if for all $k \in [K]$ the following two equivalent conditions are satisfied:*

$$i) \quad \sum_{a\in A} p_a\Big(\sum_{i=1}^{K} f_a^i\Big)\,(f_a^k - x_a^k) \le 0 \qquad \textit{for all feasible flows } \boldsymbol{x} \tag{3.7}$$

$$ii) \quad \sum_{a\in P} p_a\Big(\sum_{i=1}^{K} f_a^i\Big) \le \sum_{a\in Q} p_a\Big(\sum_{i=1}^{K} f_a^i\Big) \qquad \textit{for all } P, Q \in \mathcal{P}_k,\ P \textit{ flow carrying w.r.t. } \boldsymbol{f}^k. \tag{3.8}$$

Note that the only difference to the optimality conditions in Lemma 3.4 is the summation in the price function up to commodity K instead of k. This reflects the offline aspect since all demands are known.

3.3 Competitive Analysis

For a given sequence of commodities $\sigma = 1, \ldots, K$ and a solution f produced by an online algorithm ALG, we denote by ALG$(\sigma) = C(f)$ its cost. According to the notation introduced in Chapter 2, the online algorithm ALG is called (strictly) *c-competitive*, if the cost of ALG is never larger than c times the cost of an optimal offline solution. The *competitive ratio* of ALG is the infimum over all $c \geq 1$ such that ALG is c-competitive, see for instance Borodin and El-Yaniv [15] and Fiat and Woeginger [32].

Remark 3.6. If the price functions $p_a(z)$ are constant for every arc $a \in A$, the algorithm SEQ is optimal for the offline problem, i.e., its competitive ratio is 1. This holds because in this case the routing problems are independent from each other. In fact, each routing decision is just a shortest path problem with respect to the constant costs. Furthermore, the offline problem is a min-cost flow problem without capacity constraints. Hence, both problems can be solved more efficiently than in the general case.

Clearly, also in the case $K = 1$, the competitive ratio of SEQ is 1.

We start with a simple example motivating the impact of routing demands in an online fashion.

Example 3.7. Consider the network displayed in Figure 3.3. Assume that all arcs have the linear price function $p_a(z) = 2z$. From Equation (3.2) it follows that the cost on every arc a is given by $(2s_a + z)z$, where s_a is the amount of flow that is already routed on arc a.

Assume that a demand of one unit from node 1 to node 4 is revealed first. The online algorithm SEQ splits the demand evenly along the two possible paths incurring a cost of 1. Then the second demand of one unit starting in node 1 and terminating in node 2 is released. Here, only arc $(1, 2)$ can be used, leading to a cost of 2. Hence, the total cost of SEQ for these demands is 3.

For an optimal offline routing, the entire demand sequence is known. It is optimal to route $\frac{3}{4}$ of the first demand along the upper path $(1, 3, 4)$ and only $\frac{1}{4}$ along the lower path $(1, 2, 4)$, cf. Lemma 3.5. This incurs a cost of 1.25. Now the second demand leads to a cost of 1.5. Therefore, the total offline cost is 2.75, which is strictly smaller than the cost for the routing generated by SEQ.

The rationale of the optimal offline routing is to sacrifice some cost (compared to SEQ) for the first demand in favor of saving cost for the latter. This example also shows that the algorithm SEQ does not have a competitive ratio less than 1.09.

First, we show that there exists no competitive deterministic online algorithm, if neither the network nor the price functions are restricted. Therefore, we generalize the above example.

Proposition 3.8. *In general, there exists no competitive deterministic online algorithm for the* ONLINEMCRP.

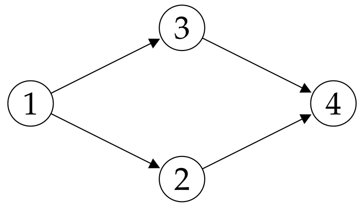

Figure 3.3: Graph construction for the proofs of Propositions 3.8, 3.33, and 3.34, and for the Example 3.7.

Proof. Consider the network depicted in Figure 3.3. For all arcs a in the network, the price function is set to $p_a(z) = m \cdot z^{m-1}$ with $m > 2$. Let $\textsc{Alg}$ be an arbitrary deterministic online algorithm. The first commodity of sequence σ has demand $d_1 = 1$ and has to be routed from node $s_1 = 1$ to node $t_1 = 4$. There are two possible paths for this commodity: path $P_1 = (1, 2, 4)$ and path $P_2 = (1, 3, 4)$. Because of symmetry, we can assume that $\textsc{Alg}$ sends a flow of $\frac{1}{2} \leq \alpha \leq 1$ over path P_1 and $(1 - \alpha)$ along path P_2. Now commodity 2 arises with demand $d_2 = 1$, source $s_2 = 1$, and target $t_2 = 2$. Algorithm $\textsc{Alg}$ has to route this demand on the only possible path $P_3 = (1, 2)$. For this sequence σ, $\textsc{Alg}$ produces a total cost of

$$\begin{aligned} \textsc{Alg}(\sigma) &= 2 \cdot \alpha^m + 2 \cdot (1 - \alpha)^m + \int_0^1 m(\alpha + z)^{m-1}\, dz \\ &= 2 \cdot \alpha^m + 2 \cdot (1 - \alpha)^m + (\alpha + 1)^m - \alpha^m. \end{aligned}$$

Routing the first commodity completely over path P_2 and the second over path P_3 leads to the total cost $2 \cdot 1^m + 1^m = 3 \geq \textsc{Opt}(\sigma)$. Letting m tend to infinity shows that in this case $\textsc{Alg}$ is not competitive. □

Despite the negative result of Proposition 3.8, we obtain competitive results in the following two sections. We first restrict the price functions to be affinely linear, then, we allow for general polynomial price functions, and finally, we study networks with a single source and a single destination.

3.3.1 Affinely Linear Price Functions

In this section, we assume that the price functions are affinely linear and show that $\textsc{Seq}$ is $\frac{4K^2}{(1+K)^2}$-competitive in this case.

For affinely linear price functions $p_a(z) = q_a \cdot z + r_a$ with $q_a \geq 0$, $r_a \geq 0$ for $a \in A$, we have for a feasible flow $(\boldsymbol{f}^1, \dots, \boldsymbol{f}^k)$

$$C_a^k(\boldsymbol{f}^k; \boldsymbol{f}^1, \dots, \boldsymbol{f}^{k-1}) = q_a \,\Big(\sum_{i=1}^{k-1} f_a^i + \tfrac{1}{2} f_a^k \Big) f_a^k + r_a\, f_a^k.$$

It follows from the optimality conditions (3.4) that if $(\boldsymbol{f}^1, \dots, \boldsymbol{f}^k)$ is generated by Seq, then

$$\sum_{a \in A} \left(q_a \sum_{i=1}^{k} f_a^i + r_a\right) (f_a^k - x_a^k) \leq 0, \tag{3.9}$$

for all feasible flows $\boldsymbol{x}^k$.

Theorem 3.9. *If the price functions are affinely linear,* Seq *is* $\frac{4K^2}{(1+K)^2}$*-competitive for the* OnlineMCRP.

Proof. We use the following useful relation at several places in the proof:

$$\sum_{k=1}^{K} \sum_{i=1}^{K} f_a^i f_a^k = 2 \sum_{k=1}^{K} \left(\sum_{i=1}^{k-1} f_a^i + \tfrac{1}{2} f_a^k\right) f_a^k. \tag{3.10}$$

Let $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K) \in \mathcal{F}_K$ be any feasible flow and let $(\boldsymbol{f}_1, \ldots, \boldsymbol{f}_K) \in \mathcal{F}_K$ be the solution produced by Seq. We start by considering the following inequality for arbitrary nonnegative real values α, β satisfying $1 \leq \alpha \leq \beta \leq 2$:

$$\begin{aligned} 0 &\leq \left(\alpha \sum_{k=1}^{K} f_a^k - \beta \sum_{k=1}^{K} x_a^k\right)^2 \\ &= \alpha^2 \sum_{k=1}^{K} \sum_{i=1}^{K} f_a^i f_a^k - 2\alpha\beta \sum_{k=1}^{K} \sum_{i=1}^{K} f_a^i x_a^k + \beta^2 \sum_{k=1}^{K} \sum_{i=1}^{K} x_a^i x_a^k. \end{aligned}$$

Using (3.10) for the first and last term, multiplying with q_a, and adding over all arcs yields:

$$\begin{aligned} 0 \leq \sum_{a \in A} q_a \Big(& 2\alpha^2 \sum_{k=1}^{K} \left(\sum_{i=1}^{k-1} f_a^i + \tfrac{1}{2} f_a^k\right) f_a^k - 2\alpha\beta \sum_{k=1}^{K} \sum_{i=1}^{K} f_a^i x_a^k + \\ &+ 2\beta^2 \sum_{k=1}^{K} \left(\sum_{i=1}^{k-1} x_a^i + \tfrac{1}{2} x_a^k\right) x_a^k \Big). \end{aligned} \tag{3.11}$$

For the next step, consider the inequality

$$\begin{aligned} 0 &\leq \sum_{a \in A} \sum_{k=1}^{K} \left((2\alpha^2 - \tfrac{2\alpha\beta}{K})\, r_a f_a^k + (2\beta^2 - 2\alpha\beta)\, r_a x_a^k\right) \\ &= \sum_{a \in A} \sum_{k=1}^{K} \left(2\alpha^2 r_a f_a^k - 2\alpha\beta\, r_a x_a^k + 2\beta^2 r_a x_a^k\right) - \tfrac{2\alpha\beta}{K} \sum_{a \in A} \sum_{k=1}^{K} r_a f_a^k. \end{aligned} \tag{3.12}$$

This inequality holds, because $K \geq 2$ and hence

$$2\alpha^2 - \tfrac{2\alpha\beta}{K} \geq 2\alpha^2 - \alpha\beta \geq 0,$$

since $1 \leq \alpha \leq \beta \leq 2$ and therefore $2\alpha - \beta \geq 0$. Furthermore, it follows that

$$2\beta^2 - 2\alpha\beta \geq 2\beta^2 - 2\beta^2 = 0.$$

Adding Inequality (3.12) to (3.11) leads to:

$$\begin{aligned}0 \leq 2\alpha^2\, C(\boldsymbol{f}) - 2\alpha\beta \sum_{a\in A}\sum_{k=1}^{K}\big(q_a \sum_{i=1}^{K} f_a^i + r_a\big)x_a^k + 2\beta^2\, C(\boldsymbol{x})\\ -\tfrac{2\alpha\beta}{K}\sum_{a\in A}\sum_{k=1}^{K} r_a\, f_a^k.\end{aligned}$$

We drop part of the second term and apply (3.9):

$$\begin{aligned}0 &\leq 2\alpha^2\, C(\boldsymbol{f}) - 2\alpha\beta \sum_{a\in A}\sum_{k=1}^{K}\big(q_a \sum_{i=1}^{k} f_a^i + r_a\big)f_a^k + 2\beta^2\, C(\boldsymbol{x})\\ &\quad -\tfrac{2\alpha\beta}{K}\sum_{a\in A}\sum_{k=1}^{K} r_a\, f_a^k\\ &= (2\alpha^2 - 2\alpha\beta)\, C(\boldsymbol{f}) - \alpha\beta \sum_{a\in A} q_a \sum_{k=1}^{K} f_a^k\, f_a^k + 2\beta^2\, C(\boldsymbol{x})\\ &\quad -\tfrac{2\alpha\beta}{K}\sum_{a\in A}\sum_{k=1}^{K} r_a\, f_a^k.\end{aligned}$$

Using the inequality of Cauchy-Schwarz and (3.10) yields:

$$\begin{aligned}0 &\leq (2\alpha^2 - 2\alpha\beta)\, C(\boldsymbol{f}) + 2\beta^2\, C(\boldsymbol{x}) - \tfrac{\alpha\beta}{K}\sum_{a\in A} q_a \big(\sum_{k=1}^{K} f_a^k\big)^2 - \tfrac{2\alpha\beta}{K}\sum_{a\in A}\sum_{k=1}^{K} r_a\, f_a^k\\ &= (2\alpha^2 - 2\alpha\beta)\, C(\boldsymbol{f}) + 2\beta^2\, C(\boldsymbol{x}) - \tfrac{2\alpha\beta}{K}\, C(\boldsymbol{f}).\end{aligned}$$

This is equivalent to:

$$C(\boldsymbol{f}) \leq \frac{\beta^2}{-\alpha^2 + \alpha\beta + \frac{\alpha\beta}{K}}\, C(\boldsymbol{x}).$$

We now take $\alpha = (1 + \frac{1}{K})$ and $\beta = 2$ to get $C(\boldsymbol{f}) \leq \frac{4K^2}{(1+K)^2}\, C(\boldsymbol{x})$, i.e., the desired bound (if we let $\boldsymbol{x}$ be an optimal offline solution). □

Remark 3.10. The parameters α and β in the previous proof are optimal solutions to the following minimization problem:

$$\min_{1\leq\alpha\leq\beta\leq 2} \frac{\beta^2}{-\alpha^2 + \alpha\beta + \frac{\alpha\beta}{K}}.$$

We do not know whether the result in Theorem 3.9 is tight. The best known lower bound for *any* deterministic online algorithm is the following.

Proposition 3.11. *In case of linear cost functions, no deterministic online algorithm for the* ONLINEMCRP *is c-competitive for any* $c < \frac{4}{3}$.

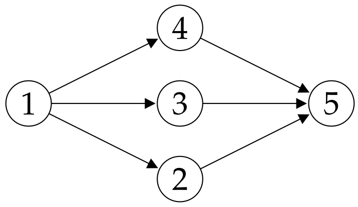

Figure 3.4: Graph construction for the proof of Proposition 3.11.

Proof. Consider the network displayed in Figure 3.4. Each arc a leaving node 1 has the same price function $p_a(z) = 4z$. All the other arcs (leading to node 5) have price function $p_a(z) = 0$. Let ALG be an arbitrary deterministic online algorithm. The first commodity with demand 1 has to be routed from $s_1 = 1$ to $t_1 = 5$.

First, assume the algorithm behaves like SEQ. This means that the demand gets evenly divided into three pieces: one third is routed over path $P_1 = (1,2,5)$, another over path $P_2 = (1,3,5)$, and the final third over path $P_3 = (1,4,5)$ (compare Lemma 3.4). Then, we reveal commodity 2 with demand 1 between nodes 1 and 2. The algorithm ALG has to route this demand on the only possible path $P_4 = (1,2)$. Therefore, the cost of ALG for this sequence σ is:

$$\textsc{Alg}(\sigma) = \textsc{Seq}(\sigma) = 3 \cdot 4 \cdot \left(\tfrac{1}{2} \cdot \tfrac{1}{3}\right) \cdot \tfrac{1}{3} + 4 \cdot \left(\tfrac{1}{3} + \tfrac{1}{2} \cdot 1\right) \cdot 1 = 4,$$

An optimal offline solution is to route half of commodity 1 over path P_2, the other half over path P_3, and commodity 2 along P_4 (compare Lemma 3.5). Therefore,

$$\textsc{Opt}(\sigma) = 2 \cdot 4 \cdot \left(\tfrac{1}{2} \cdot \tfrac{1}{2}\right) \cdot \tfrac{1}{2} + 4 \cdot \left(\tfrac{1}{2} \cdot 1\right) \cdot 1 = 3.$$

This leads to

$$\frac{\textsc{Alg}(\sigma)}{\textsc{Opt}(\sigma)} = \frac{4}{3}.$$

If ALG does not behave like SEQ for the first commodity, we can assume by symmetry that ALG routes a flow of $\alpha > \frac{1}{3}$ over path P_1. Hence, a demand of $1 - \alpha$ is routed over path P_2 and P_3. The best way to do this is to route $(1-\alpha)/2$ over each path. Then commodity 2 is released as above, again leaving no routing choice. The cost of ALG for this sequence σ is

$$\textsc{Alg}(\sigma) \geq 4 \cdot \left(\tfrac{1}{2} \cdot \alpha\right) \cdot \alpha + 2 \cdot 4 \cdot \left(\tfrac{1}{2} \cdot \tfrac{(1-\alpha)}{2}\right) \cdot \tfrac{(1-\alpha)}{2} + 4 \cdot \left(\alpha + \tfrac{1}{2} \cdot 1\right) \cdot 1 > 4.$$

since $\alpha > \frac{1}{3}$. Because $\textsc{Opt}(\sigma) = 3$, we have

$$\frac{\textsc{Alg}(\sigma)}{\textsc{Opt}(\sigma)} > \frac{4}{3}.$$

Therefore, ALG cannot have a competitive ratio less than $\frac{4}{3}$. □

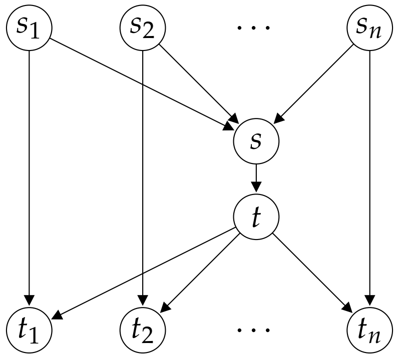

Figure 3.5: Graph construction for the proof of Proposition 3.12.

The following proposition provides an improved lower bound for SEQ.

Proposition 3.12. *In case of affine linear cost functions the online algorithm* SEQ *for* ONLINEMCRP *has a competitive ratio greater or equal to* $\frac{2K-1}{K}$.

Proof. Consider the network shown in Figure 3.5 with the following price functions: $p_{(s_i,s)}(z) = 0$, $p_{(t,t_i)}(z) = 0$, $p_{(s_i,t_i)}(z) = i$, and $p_{(s,t)}(z) = z$, for $i = 1, \ldots, n$. For $i = 1, \ldots, n$ we consecutively release a demand of size 1 from s_i to t_i. Using Lemma 3.4, we see that SEQ routes every demand over arc (s,t). The cost for these n demands is:

$$\tfrac{1}{2} \cdot 1 + (1 + \tfrac{1}{2}) \cdot 1 + \cdots + (n - 1 + \tfrac{1}{2}) \cdot 1 = \sum_{i=1}^{n} \tfrac{2i-1}{2} = \tfrac{1}{2} n^2.$$

The $(n+1)$-st demand of size $d \geq 1$ is released from s to t and incurs the following cost:

$$(n + \tfrac{1}{2}d)d = nd + \tfrac{1}{2}d^2.$$

Thus, the total cost for SEQ is given by:

$$\text{SEQ}(\sigma) = \tfrac{1}{2}(n^2 + 2nd + d^2).$$

In an optimal offline solution the first n demands are routed along the arcs (s_i, t_i) and the last demand is routed on (s,t). Hence, the total cost is:

$$\text{OPT}(\sigma) = \sum_{i=1}^{n} i + \tfrac{1}{2}d^2 = \tfrac{n(n+1)}{2} + \tfrac{1}{2}d^2.$$

Setting $d = n + 1$ and substituting $n = K - 1$ yields

$$\frac{\text{SEQ}(\sigma)}{\text{OPT}(\sigma)} = \frac{n^2 + 2nd + d^2}{n^2 + n + d^2} = \frac{1 + 2n}{1 + n} = \frac{2K - 1}{K},$$

which proves the theorem. □

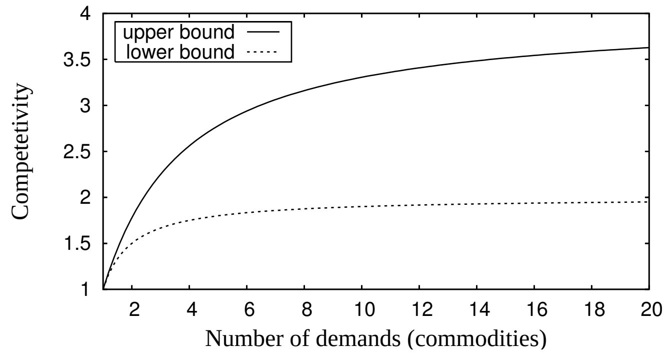

Figure 3.6: Upper bound $\frac{4K^2}{(K+1)^2}$ versus lower bound $\frac{2K-1}{K}$ on the competitive ratio of SEQ for affine linear price functions.

Remark 3.13. The value $d = n + 1$ solves the following optimization problem with respect to d:

$$\max_{d \geq 1} \frac{n^2 + 2nd + d^2}{n^2 + n + d^2} = \frac{1 + 2n}{1 + n}.$$

This yields the best lower bound for the network in the proof of Proposition 3.12.

Figure 3.6 illustrates the lower and upper bounds on the competitive ratio of the online algorithm SEQ for affine linear price functions. The bounds asymptotically converge to 2 and 4, respectively, for K tending to infinity.

A characteristic of SEQ is that it splits demand and distributes it onto several paths. We close this section by showing that only algorithms that split demand can be competitive.

Proposition 3.14. *Every deterministic online algorithm for the* ONLINEMCRP *that routes all demands unsplittably is not competitive, even for linear cost functions.*

Proof. Consider the network shown in Figure 3.7. This network contains $n + 2$ nodes and n paths from node s to node t. The price functions are $p_a(z) = 2z$ for all arcs a. Let ALG be an arbitrary deterministic online algorithm that does not split demands. We consider a single commodity with demand 1 between

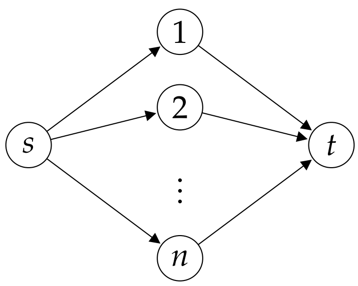

Figure 3.7: Graph construction for the proof of Proposition 3.14.

nodes s and t. Since ALG does not split, the cost of its routing is independent from the chosen path:

$$\text{ALG}(\sigma) = 2 \cdot (\tfrac{1}{2} \cdot 1) \cdot 1 + 2 \cdot (\tfrac{1}{2} \cdot 1) \cdot 1 = 2.$$

An optimal solution splits the demand into n evenly divided pieces and sends each piece over a different path. This leads to an optimal cost of

$$\text{OPT}(\sigma) = n\big(2 \cdot (\tfrac{1}{2} \cdot \tfrac{1}{n}) \cdot \tfrac{1}{n} + 2 \cdot (\tfrac{1}{2} \cdot \tfrac{1}{n}) \cdot \tfrac{1}{n}\big) = n \cdot 2 \cdot (\tfrac{1}{n})^2 = \tfrac{2}{n}.$$

Therefore, the competitive ratio of ALG is not smaller than n. Since this holds for all $n \in \mathbb{N}$, ALG is not competitive. □

3.3.2 General Price Functions

In this section, we extend Theorem 3.9 to allow for general nondecreasing price functions. Before we start with the technical exposition, we motivate the approach. For every commodity k, we use the variational inequality (3.4). Summing this inequality over $k \in [K]$ yields an inequality in terms of the entire flow $\boldsymbol{f}$ and an arbitrary feasible flow $\boldsymbol{x}$. Then, the challenge is to associate part of this expression with the total cost of $\boldsymbol{f}$ and the remaining part with the total cost of $\boldsymbol{x}$.

Definition 3.15
For a given sequence of commodities σ and a flow $\boldsymbol{f}$ that is produced by SEQ, we define

$$V^i(\boldsymbol{f}^1, \dots, \boldsymbol{f}^i, \boldsymbol{x}^i) := \sum_{a \in A} p_a\Big(\sum_{k=1}^{i} f_a^k\Big)\,(x_a^i - f_a^i)$$
$$V(\boldsymbol{f}, \boldsymbol{x}, K) := \sum_{i=1}^{K} V_i(\boldsymbol{f}^1, \dots, \boldsymbol{f}^i, \boldsymbol{x}^i),$$

where $\boldsymbol{x}^1, \dots, \boldsymbol{x}^K \in \mathcal{F}_K$ is any feasible flow.

Lemma 3.16. *A feasible flow $\boldsymbol{f}$ for a sequence σ that is produced by* SEQ *satisfies:*

$$V(\boldsymbol{f}, \boldsymbol{x}, K) \geq 0, \quad \textit{for all feasible flows } \boldsymbol{x} \textit{ for } \sigma.$$

Furthermore,

$$V(\boldsymbol{f}, \boldsymbol{x}, K) = \sum_{a \in A} V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, K),$$

where $V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, K)$ is defined as

$$V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, K) := \sum_{i=1}^{K} p_a\Big(\sum_{k=1}^{i} f_a^k\Big)\,(x_a^i - f_a^i).$$

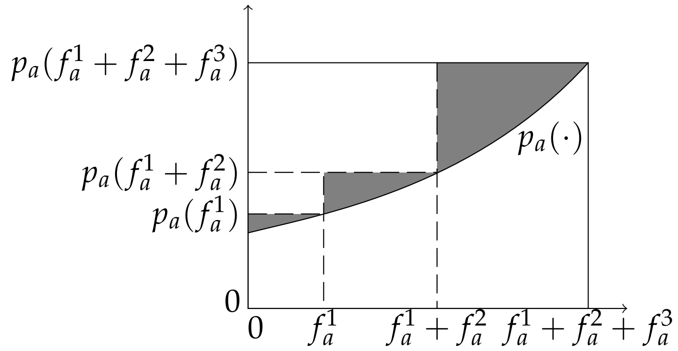

Figure 3.8: Illustration of the value $\vartheta_a^K(p_a, \boldsymbol{f})$ in the Definition 3.17 for $K = 3$. The shaded area corresponds to the value $-\vartheta_a^K(p_a, \boldsymbol{f})$.

Proof. From Lemma 3.4 we know that $V^i(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^i, \boldsymbol{x}^i)$ is nonnegative for all $i = 1, \ldots, K$. Summing over i proves the first claim. The second claim follows by changing the summation order. □

Definition 3.17
For a feasible flow $\boldsymbol{f} \in \mathcal{F}_K$, we define

$$\vartheta_a^K(p_a, \boldsymbol{f}_a) := \int_0^{f_a} p_a(z)\, dz - \sum_{i=1}^{K} p_a\Big(\sum_{k=1}^{i} f_a^k\Big)\, f_a^i.$$

Remark 3.18. For nondecreasing price functions, the value $\vartheta_a^K(p_a, \boldsymbol{f}_a)$ is non-positive for any feasible flow $\boldsymbol{f}$. The value captures the difference between the exact integral over f_a with respect to $p_a(\cdot)$ and the right-hand Riemann sum, which is greater than or equal to the integral provided nondecreasing price functions are considered. See Figure 3.8 for a graphical depiction of this value.

In the following, we reduce the entire analysis to considering the cost on a single arc. Then, by taking the supremum over all arcs, the results carry over to the general case. We define for every $a \in A$, nonnegative vectors $\boldsymbol{f}_a, \boldsymbol{x}_a \in \mathbb{R}^K$, and nonnegative real number $\lambda \geq 0$ the following values:

$$\omega(p_a; K, \lambda) := \sup_{\boldsymbol{f}_a, \boldsymbol{x}_a \geq 0} \left[\frac{\big(p_a(f_a) - \lambda\, p_a(x_a)\big)\, x_a + \vartheta_a^K(p_a, \boldsymbol{f}_a)}{p_a(f_a)\, f_a} \right], \tag{3.13}$$

$$\delta(p_a) := \sup_{f_a \geq 0} \left[p_a(f_a)\, f_a \Big(\int_0^{f_a} p_a(z) dz \Big)^{-1} \right]. \tag{3.14}$$

We assume $0/0 = 0$ by convention. For a given class $\mathcal{C}$ of nondecreasing price functions, we further define

$$\omega(\mathcal{C}; K, \lambda) := \sup_{p_a \in \mathcal{C}} \omega(p_a; K, \lambda), \quad \delta(\mathcal{C}) := \sup_{p_a \in \mathcal{C}} \delta(p_a).$$

Note that a similar value $\beta(\mathcal{C})$ without the term $\vartheta_a^K(p_a, \boldsymbol{f}_a)$ and with $\lambda = 1$ was first defined in Correa, Schulz, and Stier-Moses [24] and also, similarly, by Roughgarden in [84] with the relation $\alpha(\mathcal{C}) = \big(1 - \omega(\mathcal{C}; K, \lambda)\big)^{-1}$. For a detailed discussion about the differences between these similar approaches, we refer to Section 4.3.5 in Chapter 4 .

We define the following feasible set for the parameter λ.

Definition 3.19 (Feasible Scaling Set)
The feasible scaling set for λ is defined as

$$\Lambda := \left\{\lambda \in \mathbb{R}^+ \middle| \big(1 - \delta(\mathcal{C})\, \omega(\mathcal{C}; K, \lambda)\big) > 0\right\}.$$

Equipped with these rather technical definitions, we present our main result.

Theorem 3.20. *Let $\boldsymbol{f} \in \mathcal{F}_K$ be a flow generated by* Seq*. Then, the competitive ratio of the online algorithm* Seq *is at most*

$$\inf_{\lambda \in \Lambda} \left(\lambda\, \delta(\mathcal{C})\, (1 - \delta(\mathcal{C})\, \omega(\mathcal{C}; K, \lambda))^{-1}\right)$$

for the OnlineMCRP*.*

Proof. Let $\boldsymbol{x} \in \mathcal{F}_K$ be any feasible flow for OnlineMCRP. Then, the following inequalities hold:

$$\begin{aligned}
C(\boldsymbol{f}) &= C(\boldsymbol{f}) + \sum_{a \in A} [\lambda\, p_a(x_a)\, x_a - \lambda\, p_a(x_a)\, x_a] \\
&\le C(\boldsymbol{f}) + \sum_{a \in A} [\lambda\, p_a(x_a)\, x_a - \lambda\, p_a(x_a)\, x_a] + V(\boldsymbol{f}, \boldsymbol{x}, K) && (3.15)\\
&\le \sum_{a \in A} [\lambda\, p_a(x_a)\, x_a + \big(p_a(f_a) - \lambda\, p_a(x_a)\big)\, x_a + \vartheta_a^K(p_a, \boldsymbol{f}_a)] && (3.16)\\
&\le \sum_{a \in A} [\lambda\, p_a(x_a)\, x_a + \omega(\mathcal{C}; K, \lambda)\, p_a(f_a)\, f_a],\\
&\le \lambda\, \delta(\mathcal{C})\, C(\boldsymbol{x}) + \delta(\mathcal{C})\, \omega(\mathcal{C}; K, \lambda)\, C(\boldsymbol{f}),
\end{aligned}$$

where inequality (3.15) follows from Lemma 3.16. Inequality (3.16) follows since the considered price functions are nondecreasing. The last two inequalities follow from the definition of $\omega(\mathcal{C}; K, \lambda)$ and $\delta(\mathcal{C})$, respectively. □

Whenever $\Lambda = \emptyset$, the above approach does not provide a bound on the competitive ratio of Seq.

In the following, we consider the class $\mathcal{C}_d$ of polynomials with nonnegative coefficients and degree at most $d \in \mathbb{N}$:

$$\mathcal{C}_d := \{a_d\, x^d + \cdots + a_1\, x + a_0 \,:\, a_s \ge 0, s = 0, \ldots, d\}.$$

Note that polynomials in $\mathcal{C}_d$ are nonnegative for nonnegative arguments, nondecreasing, and convex. We first derive a bound on the value $\delta(\mathcal{C}_d)$, depending on d.

Lemma 3.21. *If the price functions of the* OnlineMCRP *are in $\mathcal{C}_d$, the value $\delta(\mathcal{C}_d)$ is at most $d+1$.*

Proof. We start with the definition of the value $\delta(p_a)$ for polynomials in $\mathcal{C}_d$.

$$\begin{aligned}\delta(p_a) &= \sup_{f_a \geq 0} \Big[\big(\sum_{i=0}^{d} a_i (f_a)^{i+1} \big) \big(\sum_{i=0}^{d} \frac{a_i}{i+1} (f_a)^{i+1} \big)^{-1} \Big] \\ &\leq \sup_{f_a \geq 0} \Big[\big(\sum_{i=0}^{d} a_i (f_a)^{i+1} \big) \big(\sum_{i=0}^{d} \frac{a_i}{d+1} (f_a)^{i+1} \big)^{-1} \Big] \\ &= 1 \big(\frac{1}{d+1} \big)^{-1} = d+1,\end{aligned}$$

where the second inequality follows since $a_i \geq 0$ and $f_a \geq 0$. ☐

Lemma 3.22. *If the price functions of the* OnlineMCRP *are in $\mathcal{C}_d$ and $\lambda \geq 1$, then, the value $\omega(\mathcal{C}_d; K, \lambda)$ is at most $\max_{0 \leq \mu} \mu - \lambda\, \mu^{d+1}$.*

Proof. By Remark 3.18, we have

$$\begin{aligned}\omega(p_a; K, \lambda) = &\sup_{f_a, x_a \geq 0} \frac{\big(p_a(f_a) - \lambda\, p_a(x_a)\big)\, x_a + \vartheta_a^K(p_a, \boldsymbol{f}_a)}{p_a(f_a)\, f_a} \\ \leq &\sup_{f_a, x_a \geq 0} \frac{\big(p_a(f_a) - \lambda\, p_a(x_a)\big)\, x_a}{p_a(f_a)\, f_a}.\end{aligned} \tag{3.17}$$

Defining

$$\mu := \begin{cases} \frac{x_a}{f_a}, & \text{for } f_a > 0 \\ 0, & \text{else,} \end{cases}$$

we have to solve

$$\max_{0 \leq \mu} \frac{\big(p_a(f_a) - \lambda\, p_a(\mu\, f_a)\big)\, \mu\, f_a}{p_a(f_a)\, f_a}$$

to bound $\omega(p_a; K, \lambda)$ from above. Without loss of generality, we can reduce the analysis to monomial price functions $p_a(x) = a_j\, x^j$ of degree $j \leq d$. The reason is that we can subdivide each arc a in d arcs a_j, $0 \leq j \leq d$ with monomial price functions $p_j(x) = a_j\, x^j$ for every arc, see Figure 3.9 for a graphical illustration. Consider now the monomial price function $p_a(x) = a_j\, x^j$ of degree j. To bound the value $\omega(p_a; K, \lambda)$ from above, we have to solve:

$$\max_{0 \leq \mu} \frac{\big(a_j\, f_a^j - \lambda\, a_j\, \mu^j\, f_a^j\big)\big)\, \mu\, f_a}{a_j\, f_a^{j+1}} = \max_{0 \leq \mu} \mu - \lambda\, \mu^{j+1}. \tag{3.18}$$

The condition $\lambda \geq 1$ implies that $\mu \leq 1$ for an optimal solution in (3.18). Thus, it is easy to see that for lower degrees $j < d$, the optimal value becomes smaller. ☐

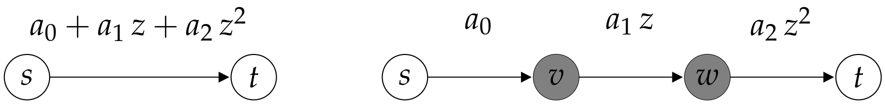

Figure 3.9: Reduction of polynomials to monomials for the proof in Lemma 3.22. By introducing the two nodes v and w, the arc (s,t) is partitioned into three separate arcs with monomial price functions.

For polynomials in $\mathcal{C}_d$ and a proper choice of λ, we can prove a bound on the value $\omega(\mathcal{C}_d; K, \lambda)$.

Proposition 3.23. *For price functions in $\mathcal{C}_d$ and $\lambda := (d+1)^{(d-1)} \geq 1$, the value $\omega(\mathcal{C}_d; K, \lambda)$ is at most $\frac{d}{(d+1)^2}$.*

Proof. We start with equation (3.18) given in the proof from Proposition 3.22 and using the highest degree d.

$$\omega(\mathcal{C}_d; K, \lambda) \leq \max_{0 \leq \mu} \mu - \lambda\, \mu^{d+1} = \max_{0 \leq \mu} \mu - (d+1)^{(d-1)}\, \mu^{d+1}.$$

The unique solution is given by $\mu^* = \frac{1}{d+1}$. Evaluating the objective leads to:

$$\omega(p_a, K; \lambda) \leq \frac{1}{d+1} - (d+1)^{(d-1)}\, (\frac{1}{d+1})^{d+1} = \frac{d}{(d+1)^2}.$$

This proves the claim. □

With the above prerequisites we can prove a constant factor bound on the competitive ratio that depends on the degree d of the considered polynomials.

Theorem 3.24. *If the price functions of the* ONLINEMCRP *are in $\mathcal{C}_d$, then, the online algorithm* SEQ *is $(d+1)^{d+1}$ -competitive.*

Proof. Let the flow $\boldsymbol{f}$ be produced by the online algorithm SEQ and let $\boldsymbol{x}$ be an arbitrary feasible flow for the ONLINEMCRP. We define $\lambda := (d+1)^{(d-1)}$ and apply Proposition 3.23, which yields $\omega(\mathcal{C}_d; K, \lambda) \leq \frac{d}{(d+1)^2}$. In order to apply Theorem 3.20, we have to verify that $\lambda \in \Lambda$. What remains to be shown is that

$$1 - \frac{(d+1)\, d}{(d+1)^2} > 0$$

holds, where the value $\delta(\mathcal{C}_d)$ is replaced by $d+1$. This inequality is equivalent to

$$\frac{1}{d+1} > 0,$$

which is trivially true. Then, applying Theorem 3.20 yields

$$C(\boldsymbol{f}) \leq \frac{(d+1)^{d-1}\,(d+1)}{(1-(d+1)\,\frac{d}{(d+1)^2})}\, C(\boldsymbol{x}) = (d+1)^{d+1}\, C(\boldsymbol{x}).$$

Taking $\boldsymbol{x}$ as the optimal offline solution proves the claim. □

The above theorem incorporates Theorem 3.9 as a special case. Note that we also get a bound of 4 for degree 1 polynomials. In Theorem 3.9, however, we incorporated the value $\vartheta_a^K(p_a, f_a)$ in the analysis giving slightly better bounds that depend on the number of commodities.

3.3.3 Lower Bounds for Polynomial Price Functions

In this section, we derive lower bounds for price functions in $\mathcal{C}_d$. Consider the network presented in Figure 3.5 with the following price functions: $p_{(s_i,s)}(z) = 0$, $p_{(t,t_i)}(z) = 0$, $p_{(s_i,t_i)}(z) = i^d$, $i = 1,\ldots,k$, and $p_{(s,t)}(z) = z^d, d \in \mathbb{N}$. We consecutively release demands of size 1 from s_i to t_i, for $i = 1,\ldots,k$. Due to the choice of the affine terms i^d, SEQ routes every demand over the arc from s to t. The cost for these k demands is:

$$\sum_{i=1}^{k} \frac{1}{d+1}\big((i-1)+1\big)^{d+1} - \frac{1}{d+1}(i-1)^{d+1} = \frac{1}{d+1}k^{d+1}.$$

Then, we release the $(k+1)$-th commodity with demand x from s to t, which generates the following cost:

$$\frac{1}{d+1}\big(k+x\big)^{d+1} - \frac{1}{d+1}(k)^{d+1}.$$

Thus, the total cost for SEQ is given by:

$$\text{SEQ}(\sigma) = \int_0^{k+x} p_{(s,t)}(z)\, dz = \frac{1}{d+1}\big(k+x\big)^{d+1}.$$

The optimal offline algorithm OPT routes the first k demands along the direct arcs from s_i to t_i incurring cost of:

$$\sum_{i=1}^{k} i^d.$$

The last demand is routed from s to t with cost $(\frac{1}{d+1})x^{d+1}$. The total cost for OPT is given by:

$$\text{OPT}(\sigma) = \sum_{i=1}^{k} i^d + \frac{1}{d+1}x^{d+1}.$$

In order to evalute the ratio of the cost of SEQ and OPT, respectively, we need the following lemma.

Lemma 3.25. *The sum of the d-th power of numbers from* 1 *to k is a polynomial in k given by:*

$$\sum_{i=1}^{k} i^d = \frac{1}{d+1}\sum_{j=0}^{d+1}\binom{d+1}{j} B_j\, k^{d+1-j},$$

where B_j are the Bernoulli numbers.

A proof for this lemma can be found in Graham, Knuth, and Patashnik [47, Ch. 7].

Theorem 3.26. *In case of price functions in $\mathcal{C}_d$, the online algorithm* Seq *for* OnlineMCRP *has a competitive ratio greater than or equal to 2^d.*

Proof. We have to show that the competitive ratio fulfills:

$$\frac{\textsc{Seq}(\sigma)}{\textsc{Opt}(\sigma)} \geq 2^d.$$

We follow the construction of the above discussion,

$$\frac{\textsc{Seq}(\sigma)}{\textsc{Opt}(\sigma)} \geq \lim_{k\to\infty} \frac{\big(k+x\big)^{d+1}}{(d+1)\sum\limits_{i=1}^{k} i^d + x^{d+1}}.$$

We set $x = k$ which yields:

$$\begin{aligned}\frac{\textsc{Seq}(\sigma)}{\textsc{Opt}(\sigma)} &\geq \lim_{k\to\infty} \frac{\big(2k\big)^{d+1}}{(d+1)\sum\limits_{i=1}^{k} i^d + k^{d+1}} \\ &= \lim_{k\to\infty} \frac{\big(2k\big)^{d+1}}{2k^{d+1} + \sum\limits_{j=1}^{d+1} \binom{d+1}{j} B_j\, k^{d+1-j}} = 2^d,\end{aligned}$$

where the equality follows from Lemma 3.25 and the fact that the highest degree coefficient is $B_0 = 1$.

□

3.4 Single Commodity Networks

Restricting the considered networks to only contain a single source and a single destination, i.e., $s_1 = \cdots = s_k$ and $t_1 = \cdots = t_k$, we can show that Seq returns an optimal solution. To get insight into the techniques required to prove this result, we start with the simpler case of a network consisting of two nodes connected by parallel arcs.

3.4.1 Parallel Arcs

We now consider the *parallel arc case*, i.e., D consists of two nodes and parallel arcs only. Recall from Lemma 3.4 and 3.5 that a flow $\boldsymbol{x}$ solves the offline problem (3.6) and $\boldsymbol{f}$ is generated by Seq if and only if for all $a, \hat{a} \in A$ and $k \in [K]$ follows

- $\sum\limits_{k=1}^{K} x_a^k > 0 \Rightarrow p_a\big(\sum\limits_{i=1}^{K} x_a^i\big) \leq p_{\hat{a}}\big(\sum\limits_{i=1}^{K} x_{\hat{a}}^i\big)$

- $f_a^k > 0 \Rightarrow p_a(\sum_{i=1}^{k} f_a^i) \leq p_{\hat{a}}(\sum_{i=1}^{k} f_{\hat{a}}^i)$.

Lemma 3.27. *Given a sequence $\sigma = 1, \ldots, K$ and let $\boldsymbol{f}$ be the flow generated by* Seq*. Define*

$$A_k^+ := \{a \in A \,:\, f_a^k > 0\}$$

for $k \in [K]$. Then,

$$p_a(\sum_{i=1}^{k+1} f_a^i) \leq p_{\hat{a}}(\sum_{i=1}^{k+1} f_{\hat{a}}^i),$$

for all $a \in A_k^+$, $\hat{a} \in A$, and $k = 1, \ldots, K-1$.

Proof. Let $a \in A_k^+$. First assume that $a \in A_{k+1}^+$. Then by the optimality conditions from above for $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^{k+1})$ the claim follows.

Now assume $a \notin A_{k+1}^+$. Then we have for all $\hat{a} \in A$:

$$\begin{aligned} p_a(\sum_{i=1}^{k+1} f_a^i) &= p_a(\sum_{i=1}^{k} f_a^i) \\ &\leq p_{\hat{a}}(\sum_{i=1}^{k} f_{\hat{a}}^i) \\ &\leq p_{\hat{a}}(\sum_{k=1}^{k+1} f_{\hat{a}}^i). \end{aligned}$$

The first inequality follows from the optimality conditions for the first k flows $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^k)$, and the second comes from the assumption that the price functions are nondecreasing. □

Proposition 3.28. *Given a sequence of commodities and let $\boldsymbol{f}$ be the flow generated by* Seq *for this sequence. Then, $C(\boldsymbol{f}) \leq C(\boldsymbol{x})$ for any feasible $\boldsymbol{x}$, i.e., $\boldsymbol{f}$ is also an offline optimum.*

Proof. For the last commodity K we have the following optimality condition:

$$p_a(\sum_{i=1}^{K} f_a^i) \leq p_{\hat{a}}(\sum_{k=1}^{K} f_{\hat{a}}^i), \tag{3.19}$$

for all $a \in A_K^+$ and $\hat{a} \in A$. Using Lemma 3.27 for $k = K - 1$ we obtain:

$$p_a(\sum_{i=1}^{K} f_a^i) \leq p_{\hat{a}}(\sum_{k=1}^{K} f_{\hat{a}}^i),$$

for all $a \in A_{K-1}^+$ and $\hat{a} \in A$. Inequality (3.19) and applying Lemma 3.27 iteratively $K - 1$ times yields the optimality conditions (3.5) for the offline optimum. □

3.4.2 Arbitrary Digraph

We know allow for an arbitrary digraph between a single source node s and a single destination node t. For this more general setting, we show that SEQ computes an optimal solution.

Theorem 3.29. *Consider an instance of the* ONLINEMCRP*, where all commodities share the same source s and destination t. Then,* SEQ *computes an optimal routing.*

Proof. The proof uses induction on the number of commodities K. For the case $K = 1$, the claim follows since by definition SEQ routes one commodity with minimum cost. Therefore, assume that the claim holds for any sequence containing $K-1$ commodities.

For the sake of contradiction, further assume that the flow $\boldsymbol{f}$ that is generated by SEQ for a given sequence with K commodities is not offline optimal. Hence, this flow does not satisfy the conditions of Lemma 3.5. Therefore, there exist paths $P, Q \in \mathcal{P}$, where P is flow carrying, with

$$\sum_{a\in P} p_a\Big(\sum_{i=1}^{K} f_a^i\Big) > \sum_{a\in Q} p_a\Big(\sum_{i=1}^{K} f_a^i\Big). \tag{3.20}$$

By the induction hypothesis the routing computed by SEQ for the first $K-1$ commodities is optimal. Therefore, Lemma 3.5 holds. Inequality (3.20) is only valid if $f_P^K > 0$. To see this, assume $f_P^K = 0$. Then, it follows that P is flow carrying with respect to the first $K-1$ commodities. Invoking the optimality conditions in Lemma 3.5 for the first $K-1$ commodities (induction hypothesis) leads to:

$$\sum_{a\in P} p_a\Big(\sum_{i=1}^{K} f_a^i\Big) = \sum_{a\in P} p_a\Big(\sum_{i=1}^{K-1} f_a^i\Big) \le \sum_{a\in Q} p_a\Big(\sum_{i=1}^{K-1} f_a^i\Big) \le \sum_{a\in Q} p_a\Big(\sum_{i=1}^{K} f_a^i\Big),$$

where the last inequality follows because the price functions $p_a(\cdot)$ are nondecreasing. Since this contradicts (3.20), we have $f_P^K > 0$. In particular, $f_a^K > 0$ for all $a \in P$. For small enough $\varepsilon > 0$, we define the *nonnegative* flow $\boldsymbol{x} := (\boldsymbol{f}^1, \ldots, \boldsymbol{f}^{K-1}, \boldsymbol{x}^K)$ with

$$x_a^K := \begin{cases} f_a^K - \varepsilon & a \in P \setminus Q \\ f_a^K + \varepsilon & a \in Q \setminus P \\ f_a^K & \text{otherwise.} \end{cases}$$

By construction this flow is feasible.

We obtain for the difference of costs:

$$\begin{aligned} C(\boldsymbol{x}) - C(\boldsymbol{f}) = &\sum_{a\in P\setminus Q} \Big[\int_0^{f_a^K-\varepsilon} p_a\Big(\sum_{i=1}^{K-1} f_a^i + z\Big)\,dz - \int_0^{f_a^K} p_a\Big(\sum_{i=1}^{K-1} f_a^i + z\Big)\,dz \Big] \\ &+ \sum_{a\in Q\setminus P} \Big[\int_0^{f_a^K+\varepsilon} p_a\Big(\sum_{i=1}^{K-1} f_a^i + z\Big)\,dz - \int_0^{f_a^K} p_a\Big(\sum_{i=1}^{K-1} f_a^i + z\Big)\,dz \Big] \end{aligned}$$

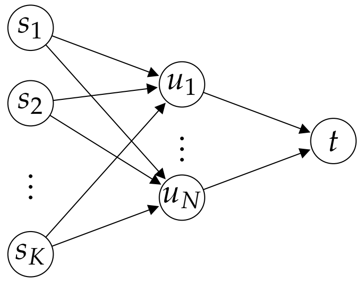

Figure 3.10: Construction for the proof of Proposition 3.30.

$$= -\sum_{a \in P \setminus Q} \int_{f_a^K - \varepsilon}^{f_a^K} p_a\big(\sum_{i=1}^{K-1} f_a^i + z\big)\,dz + \sum_{a \in Q \setminus P} \int_{f_a^K}^{f_a^K + \varepsilon} p_a\big(\sum_{i=1}^{K-1} f_a^i + z\big)\,dz$$

$$= \int_0^{\varepsilon} \Big(- \sum_{a \in P \setminus Q} p_a\big(\sum_{i=1}^{K} f_a^i + z - \varepsilon\big) + \sum_{a \in Q \setminus P} p_a\big(\sum_{i=1}^{K} f_a^i + z\big) \Big)\,dz. \tag{3.21}$$

We now define

$$g(z,\varepsilon) := -\sum_{a \in P \setminus Q} p_a\big(\sum_{i=1}^{K} f_a^i + z - \varepsilon\big) + \sum_{a \in Q \setminus P} p_a\big(\sum_{i=1}^{K} f_a^i + z\big).$$

By (3.20) we have $g(0,0) < 0$. Since $p_a(\cdot)$ is continuous, g is continuous, too. Hence, $g(z,\varepsilon) < 0$ for all z and δ with $0 \leq z, \varepsilon < \delta$, if δ is small enough. Therefore, the right hand side of (3.21) is strictly smaller than 0. It follows that $C(\boldsymbol{x}) < C(\boldsymbol{f})$. This is a contradiction since $\boldsymbol{x}$ and $\boldsymbol{f}$ only differ with respect to the last commodity K and SEQ solves problem (3.3). Hence, SEQ computes an offline optimal solution. □

3.5 Unsplittable Routings

In this section we study the variant of the ONLINEMCRP with unsplittable routings, i.e., the demand of each commodity has to be routed on a single path. Such a restriction often occurs in practice, for instance in single path routing problems in telecommunication networks. It is possible to formulate a mixed integer convex program for this setting. In contrast to the splittable case, however, the offline problem is NP-hard.

Proposition 3.30. *The offline problem for the* ONLINEMCRP *with unsplittable routings is NP-hard, even when the price functions are linear.*

Proof. Consider an instance of the *minimum sum of squares problem*, which is known to be NP-complete in the strong sense (see Garey and Johnson [43]). Here, are given nonnegative integers $d_1, \ldots, d_K$ and positive integers $N \leq K$

and J. The question is if there exists a partition of $[K]$ into N sets $A_1, \dots, A_N$ such that

$$\sum_{i=1}^{N} \Big(\sum_{k \in A_i} d_k \Big)^2 \leq J?$$

For the reduction to the offline problem, we construct a network D with node set $\{s_1, \dots, s_K, u_1, \dots, u_N, t\}$ and the following arcs: For each $k \in [K]$ and $i \in [N]$ we have an arc (s_k, u_i) with price function 0. For each $i \in [N]$ we add an arc $a = (u_i, t)$ with price function $p_a(z) = 2\,z$; see Figure 3.10. Furthermore, for $k \in [K]$ there are demands d_k between s_k and t.

We claim that there exists an unsplittable solution to the offline problem of value at most J if and only if the answer to the minimum sum of squares problem is positive. To see this, first assume that $A_1, \dots, A_N$ is the wanted partition. Then if $k \in A_i$, we route commodity k along u_i to t. Using (3.10), we obtain the following costs:

$$2 \sum_{i=1}^{N} \sum_{k \in A_i} \Big(\sum_{\substack{j \in A_i \\ j<k}} d_j + \frac{1}{2} d_k \Big) d_k = \sum_{i=1}^{N} \sum_{k \in A_i} \sum_{j \in A_i} d_k\, d_j = \sum_{i=1}^{N} \Big(\sum_{k \in A_i} d_k \Big)^2.$$

This proves the forward direction of the claim. Conversely, assume that there exists an unsplittable flow of value J. For $i = 1, \dots, N$, let A_i be the set of indices k whose corresponding demands are routed over the arc (u_i, t). Again the cost is given as above, which shows that there exits a solution to the minimum sum of squares problem. □

Remark 3.31. The unsplittable variant of (3.3) can be computed in polynomial time since it amounts to solving a shortest path problem.

To see this, consider the set of arcs within the path system $\mathcal{P}_k$ of commodity k. For this set, we evaluate the arc cost $C_a^k(d_k; f_a^1, \dots, f_a^{k-1})$ with respect to the demand d_k. Defining these values as routing weights, the solution of the unspittable variant of (3.3) amounts to a shortest path problem with respect to the routing weights.

Remark 3.32. When the price functions are constant, both the unsplittable variants of (3.3) and (3.6) can be written as (integer) min-cost flow problems. Hence, they can be solved in polynomial time, see e.g. Schrijver [85, Ch. 12].

The following two results show that the additional requirement of unsplittable routings does not improve competitiveness properties of the OnlineMCRP. The first is the unsplittable version of Proposition 3.8.

Proposition 3.33. *In general there exists no competitive deterministic online algorithm for the unsplittable variant of the* OnlineMCRP.

Proof. Consider again the network shown in Figure 3.3, where each arc a has the price function $p_a(z) = m \cdot z^{m-1}$ for some $m > 2$. Let Alg be an arbitrary deterministic online algorithm. We first reveal a commodity with demand $d_1 = 1$, source $s_1 = 1$, and target $t_1 = 4$. Without loss of generality, we can assume that ALG uses path $P_1 = (1, 2, 4)$ to route this demand. We then release commodity 2 with demand $d_2 = 1$, source $s_2 = 1$, and target $t_2 = 2$. The algorithm ALG has to route this commodity on the single path $P_2 = (1, 2)$. Hence, for this sequence σ, ALG yields the cost

$$\text{ALG}(\sigma) = 2 \cdot 1^m + \int_0^1 m(1+z)^{m-1}\, dz = 2 + (1+1)^m - 1^m = 1 + 2^m.$$

The optimal cost is $\text{OPT}(\sigma) = 3$, which is achieved by routing commodity 1 over path $P_3 = (1, 3, 4)$ and commodity 2 along path P_2. Therefore, for m going to infinity it follows that ALG is not competitive. □

We can also improve the lower bound of Proposition 3.11 from $\frac{4}{3}$ to 2.

Proposition 3.34. *If we consider only linear price functions in $\mathcal{C}_1$, no deterministic online algorithm has a competitive ratio less than 2 for the unsplittable variant of the* ONLINEMCRP.

Proof. Consider the network shown in Figure 3.3, where each link a has the same price function $p_a(z) = 2z$. Let ALG be an arbitrary deterministic online algorithm. We first reveal commodity 1 with demand $d_1 = 1$, source $s_1 = 1$, and target $t_1 = 4$. Without loss of generality this commodity is routed over path $P_1 = (1, 2, 4)$. Then we release one commodity from node 1 to 2 and one commodity from node 2 to 4. Both have a demand of 1. Since for each of the last two commodities there exists only a single path, the assignment by ALG for this sequence σ leads to a cost of

$$\text{ALG}(\sigma) = 2 \cdot 2 \cdot (\tfrac{1}{2} \cdot 1) \cdot 1 + 2 \cdot (1 + \tfrac{1}{2} \cdot 1) \cdot 1 + 2 \cdot (1 + \tfrac{1}{2} \cdot 1) \cdot 1 = 8.$$

An optimal routing is achieved by routing commodity 1 along path $P_2 = (1, 3, 4)$ and commodity 2 and 3 over their single paths. Since the optimal cost for σ is $\text{OPT}(\sigma) = 4$, the competitive ratio of ALG is at least 2. □

3.6 Computational Study

In the previous sections, we introduced the framework ONLINEMCRP in order to analyze the efficiency of online multicommodity routing strategies for networks with nondecreasing price functions. In particular, we studied the greedy online algorithm SEQ that routes a commodity with minimum cost. The framework is based on the assumption that every demand can be split into infinitesimal small pieces that can be routed consecutively and each piece prompts an update of arc prices. In other words, the bundle size can be arbitrarily small. The derived analytical results are based on competitive analysis coming from the classical toolbox in the online optimization field. It is inherent to the concept of competitive analysis that the competitive ratio of an online

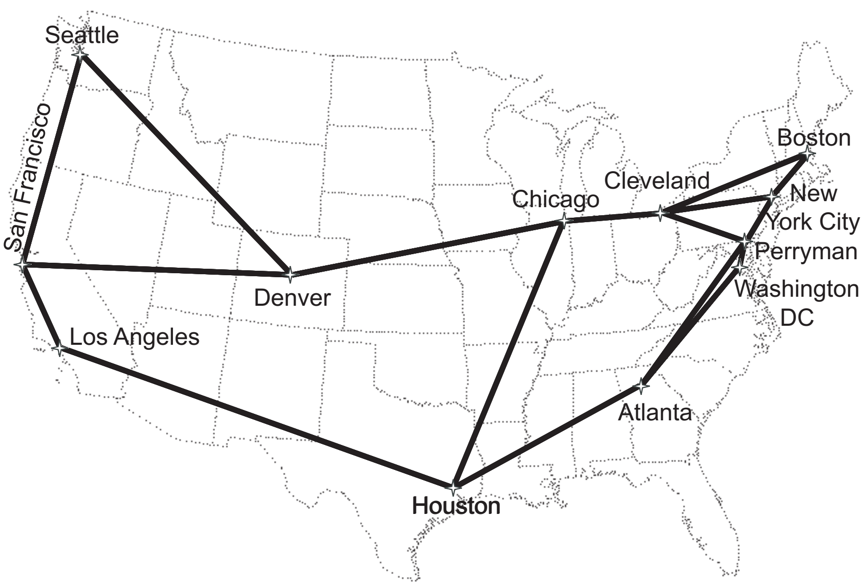

Figure 3.11: Very High Performance Backbone Network Service (vBNS) network topology. Every node represents and Internet service provider.

algorithm holds for every instance. Worst-case instances, however, maybe very rare or even not possible to construct in practice. To assess the efficiency of the routing strategies in practical environments, we present a computational study for more realistic settings.

In particular, we present an empirical case study under the following assumptions: (i) we consider the Very High Performance Backbone Network Service (vBNS) network topology that connects all major cities and universities in the US, see Figure 3.11; (ii) network arcs are equipped with finite capacities; (iii) traffic demands are generated stochastically over time; (iv) demands expire over time; (v) each demand has a discrete bundle size and is routed along a single path. In the reminder of this section, we empirically quantify the efficiency loss of a greedy online routing algorithm modeling the iREX protocol.

3.6.1 Online Routing with Expiration

As in the previous sections, we consider a set $[K] := \{1, \ldots, K\}$ of source-destination pairs that represent the inter-domain reservation requests (demands). For each $k \in [K]$, a demand of d_k must be routed from the source s_k to the destination t_k. Without loss of generality, we assume that all demands have the same normalized bundle size. A demand value that is larger than this bundle size can be represented by several demands of bundle size. We introduce a starting time τ_k that specifies the release time of commodity k. Furthermore, every demand k has a duration time E_k. Without loss of generality, we assume the time points are ordered $\tau_1 < .. < \tau_K$. We define $[K(\tau)] \subseteq [K]$ to be the subset of commodities that are active just after time τ. Formally, the set is defined as $[K(\tau)] := \{i \in [K] \,|\, \tau \in (\tau_i, \tau_i + E_i]\}$. We further define $[\bar{K}(\tau)] := \{i \in [K] \mid \tau \in [\tau_i, \tau_i + E_i]\}$ to be the set of commodities that are active at time τ. The resource arcs $a \in A$ of the network are equipped with finite resource capacities $c = (c_a, a \in A)$.

We focus in this section on the greedy online algorithm Seq with a discrete bundle size as discussed in Remark 3.1. This algorithm models the iREX protocol as specified by Yahaya and Suda in [88, 89]. According to this protocol, every commodity $k \in [K]$ is routed along the cheapest feasible path. This is equivalent to solving the following linear min cost flow problem:

$$
\begin{aligned}
(L_k) \qquad \min \quad & \sum_{a \in A} p_a \Big(\sum_{i \in [K(\tau_k)]} f_a^i \Big) f_a^k \\
\text{s.t.} \quad & \sum_{a \in \delta^+(v)} b_a^k - \sum_{a \in \delta^-(v)} b_a^k = \gamma_k(v), && \forall\, v \in V \qquad (3.22) \\
& f_a^k = d_k\, b_a^k && \forall\, a \in A \\
& \sum_{i \in [\bar{K}(\tau_k)]} f_a^i \le c_a, \qquad b_a^k \in \{0,1\}, && \forall\, a \in A,
\end{aligned}
$$

where $\delta^+(v)$ and $\delta^-(v)$ are the arcs leaving and entering v, respectively; furthermore, $\gamma_k(v) = 1$ if $v = s_k$, $\gamma_k(v) = -1$ if $v = t_k$, and $\gamma_k(v) = 0$ otherwise.

The value $\sum_{i \in [K(\tau_k)]} f_a^i$ captures the aggregated demand that is active at time τ_k. The terms $p_a\Big(\sum_{i \in [K(\tau_k)]} f_a^i\Big)$ are constant, hence L_k is a linear program. A solution is simply the cheapest feasible path for commodity k.

3.6.2 The Offline Optimum

In Section 3.2.2, we defined the optimal offline solution as the flow that minimizes total routing cost for a given sequence of commodities. Beside considering monetary cost, network congestion is also an important metric as suggested by Yahaya and Suda [88, 89]. They presented simulations with nondecreasing price functions and showed that the iREX protocol performs with less network congestion when compared to the current existing methods for deploying end-to-end inter-domain traffic with QoS requirements. [2]

[2]The current method for deploying end-to-end inter-domain traffic with QoS requirements is the Service Level Agreement (SLA) method. A detailed description of the SLA method can be found in the Frame Relay Forum [41] and in Goderis et al [86]. The SLA method is for resource consumer domains to negotiate with a neighboring resource provider domain and create a business level document called the Service Level Agreement (SLA)[41][86]. The SLA document defines the expectations and responsibilities of both the resource consumer and the resource provider domains. This includes the specifications of the QoS service negotiated for in the form of a technical document called the Service Level Specification (SLS)[46]. When an agreement is reached at the business level, the service specifications defined by the SLS are then installed as policy by the network administrator of the resource provider domain. There are two major problems with the SLA method. First, consumer domains cannot choose transit inter-domain resource provider domains beyond the first inter-domain hop. This constitutes a lack of control and can negatively impact the interest of a resource consumer. Secondly,

Our goal is to analyze the efficiency of this algorithm compared to some global optimum. Since the total traffic load varies over time, we evaluate the efficiency of the iREX protocol at different time points τ. Our measure of efficiency is again based on competitive analysis coming from online optimization. We will present two variants of an offline optimum. In the first variant, the offline optimum corresponds to a minimum cost flow, if the demands are nonexpiring and the price functions are used to model congestion on the network arcs. In this regard, all analyical results derived in the previous sections carry over to the congestion metric under the above described conditions. In the second variant, we focus on the most congested arc in the network, see Fortz and Thorup [37] for models and algorithms for minimizing congestion in telecommunication networks.

Congestion Functions

We evaluate the iREX protocol with respect to network congestion. One way to define a minimum congestion network is to assign a nondecreasing congestion function ℓ_a to each arc $a \in A$. These functions are typically nonlinear, positive, and strictly increasing with flow, see Patriksson [73]. In practical applications, the most frequently used functions are polynomials, whose degrees and coefficients are determined from real-world data through statistical evaluation methods, see Patrikkson [73] and Branston [17]. The total congestion cost for a flow $\boldsymbol{f}$ is defined as

$$\Phi(\boldsymbol{f}) = \sum_{a \in A} \ell_a(f_a)\, f_a. \tag{3.23}$$

The idea is that it will be cheap to send traffic over an underutilized arc, but as the load on the arc increases the cost for this arc will grow super linearly, penalizing high congestion. Hence, minimizing convex load dependent cost functions are well suited to balance the load in a network, see also Fortz and Thorup [37].

A flow that minimizes congestion in a network at time τ solves the following optimization problem:

$$\begin{aligned} (P_1) \qquad \min \quad & \Phi(\boldsymbol{f}) \\ \text{s.t.} \quad & \sum_{a \in \delta^+(v)} f_a^k - \sum_{a \in \delta^-(v)} f_a^k = \gamma(v) \quad \forall\, v \in V,\ k \in [\bar{K}(\tau)] \\ & f_a \le c_a, \qquad f_a^k \ge 0 \qquad\qquad \forall\, a \in A,\ k \in [\bar{K}(\tau)], \end{aligned} \tag{3.24}$$

where $\gamma(v)$ is defined as in (3.1).

The solution of problem P_1 is the *offline* optimum, where *all* demands d_k, $k \in [K(\tau)]$ that are active at time τ are taken into account. Since problem P_1 has a convex objective and linear constraints, a global optimum exists and can be computed with arbitrary precision in polynomial time, see Grötschel,

---

the manual SLA process is very slow (in the order of days). Thus, it is impossible to take advantage of some knowledge of the network state, since by the time the choice is deployed, the network state would have changed.

Lovasz and Schrijver [48]. If P_1 has a strictly convex objective, the global optimum is also unique. Note that P_1 allows for splitting demands along different paths. Incorporating binary variables $b_a^k \in \{0,1\}$ for each resource arc and demand, the condition $f_a^k = d_k\, b_a^k$ together with replacing (3.24) with

$$\sum_{a\in\delta^+(v)} b_a^k - \sum_{a\in\delta^-(v)} b_a^k = \begin{cases} 1, \text{ if } v = s_k \\ -1 \text{ if } v = t_k \\ 0, \text{ otherwise,} \end{cases}$$

accounts for single path routing. We call this type of problem SP_1.

Remark 3.35. Problem SP_1 is $\mathcal{NP}$-hard.

It is known that single path (unsplittable) multicommodity flow problems with capacities involving a linear objective are $\mathcal{NP}$-hard as shown by Kleinberg in [59]. So this holds certainly for convex objectives. However, we have the following rather trivial bounds:

Proposition 3.36. *Let* $\boldsymbol{f} = (\boldsymbol{f}^k, k \in [\bar{K}(\tau)])$ *be a feasible flow that is produced by the solutions of problem* L_k *at time* τ. *Let* $\boldsymbol{g}$ *and* $\boldsymbol{h}$ *be optimal flows of* P_1 *and* SP_1, *respectively. Then, the following inequalities are satisfied:*

$$\Phi(\boldsymbol{g}) \le \Phi(\boldsymbol{h}) \le \Phi(\boldsymbol{f}). \tag{3.25}$$

Proof. Each flow $\boldsymbol{f}^k$ routes the demand d_k on a single path. Hence, $\boldsymbol{f}$ is feasible for problem SP_1, i.e., $\Phi(\boldsymbol{h}) \le \Phi(\boldsymbol{f})$. Furthermore, $\boldsymbol{h}$ is a feasible flow for P_1. Therefore, $\Phi(\boldsymbol{g}) \le \Phi(\boldsymbol{h})$. □

To evaluate the performance of the solutions of L_k, we numerically solve P_1 and SP_1, which provides us with the lower bounds $\Phi(\boldsymbol{g}) \le \Phi(\boldsymbol{h}) \le \Phi(\boldsymbol{f})$. Furthermore, we can empirically quantify the gain of the fractional routing compared to the unsplittable variant.

Minimizing the Most Congested Arc

Another way to define a minimum congestion network is to minimize the load on the most congested arc. This leads to:

$$\begin{aligned}
(P_2) \qquad \min \quad & \Gamma(\boldsymbol{f}) = \max_{a\in A} \ell_a(f_a)\, f_a \\
\text{s.t.} \quad & \sum_{a\in\delta^+(v)} f_a^k - \sum_{a\in\delta^-(v)} f_a^k = \gamma(v) && \forall\, v \in V,\ k \in [\bar{K}(\tau)] \\
& f_a \le c_a, \qquad f_a^k \ge 0 && \forall\, a \in A,\ k \in [\bar{K}(\tau)],
\end{aligned}$$

where $\gamma(v)$ is defined as in (3.1). The drawback of this formulation is that once the bottleneck of the network (i.e. the most congested arc) is identified, the routing on remaining resource arcs does not affect the objective. This might lead to inferior routing decisions in terms of the metric used in P_1. Using the same arguments as before, we have the following bound:

Proposition 3.37. *Let $\boldsymbol{f} = (\boldsymbol{f}^k, k \in [\bar{K}(\tau)])$ be a flow produced by the solutions of L_k at time τ. Let $\boldsymbol{g}$ and $\boldsymbol{h}$ be optimal flows of P_2 and (SP_2), respectively. Then, the following inequalities are satisfied:*

$$\Gamma(\boldsymbol{g}) \leq \Gamma(\boldsymbol{h}) \leq \Gamma(\boldsymbol{f}). \tag{3.26}$$

Here problem SP_2 is the single path variant of problem P_2.

3.6.3 Numerical Results

We simulate the iREX protocol with the iREX simulator under different traffic loads, see Yahaya and Suda [88, 89] for a description of this simulator. Then, we evaluate network congestion at different times τ and compare the resulting flow of these snapshots with four lower bounds for each snapshot. These lower bounds are derived by solving the four associated offline optimization problems P_1, SP_1 and P_2, SP_2.

For completeness, we also conducted simulations with the current Service Level Agreement (SLA) method for deploying end-to-end inter-domain traffic with QoS requirements.

The Simulator

The iREX simulator implements the iREX protocol and a simplified Border Gateway Protocol (BGP). The BGP protocol is needed to simulate the SLA method. The simulator performs packet level simulation for control packets used for iREX and BGP signaling, and flow level simulation for the deployment of flows with QoS constraints.

We have used iREX simulation subconfigurations based on the type of price function used by domains to price their resources. The *linear* configuration prices resources uniformly according to the affine linear function $p(z) = a_0 + a_1 z$. The *squared* configuration prices resources uniformly according to the squared polynomial $p_a(z) = a_0 z + a_1 z + a_2 z^2$. The *random* subconfiguration randomly assigns each domain one of three price functions - linear, squared or cubed ($p(z) = a_0 z + a_1 z + a_2 z^2 + a_3 z^3$). All coefficients a_i are assumed to be nonnegative. These coefficients are randomly assigned for the *random* subconfiguration. The topology chosen for the simulations is the Very High Performance Backbone Network Service (vBNS) topology with each point of presence representing an Internet service provider (ISP) domain. ISP domains are assumed to be connected with OC48 optical fiber arcs to its neighbors and the length of each arc is calculated to be the actual beeline distance between the cities. Figure 3.11 illustrates the chosen topology. Inter-domain reservation requirements within the simulator are viewed as bundles of traffic sized 0.1% of line speed (about 2.4mb/sec) with a 5 minute average reservation duration (E_k). The traffic load (total projected bandwidth usage) is determined according to a percentage of each domain's actual total egress capacity in the topology from 0% to 100% in 4% steps. To generate demands, we used a simple Poisson arrival model with parameters derived from a $M/M/\infty$ analysis.

Mathematical Solutions

To efficiently compute solutions for all problems of type P_1, SP_1, and P_2, SP_2 we used CPLEX 10.0, that is equipped with Linear (LP), Quadratic (QP), Mixed Integer Problem (MIP), and Quadratic Integer (QIP) solvers. For modeling purposes, we used the ZIMPL modeling language, see Koch [60]. In total, more than 2600 problems of type P_1, and P_2 are solved to optimality. We solved the problems SP_1, SP_2 involving integer constraints within 1% of optimality. Average running time on a Pentium 4 (3GHz) for the problem type P_1, P_2 was about 1 second and for problem types SP_1, SP_2 about 30 seconds.

Metrics

We present efficiency results using two metrics, the efficiency loss compared to solutions of P_1 and SP_1, and compared to solutions of P_2 and SP_2. These results are from 4 simulation runs for the *linear* and *squared* sub configurations, and 16 runs for the *random* sub configurations, with individual runs having approximately 500,000 simulated reservations. To compare the simulation results with the lower bound of P_1, SP_1, and P_2, SP_2, we evaluated congestion for a simulated flow $\boldsymbol{f}$ by evaluating $\Phi(\boldsymbol{f})$ and $\Gamma(\boldsymbol{f})$ at time points τ. For all graphs, we define *efficiency loss* to be the percentage difference between the network congestion of the iREX protocol simulation results and the computed optimal solutions as defined by the problems P_1 and SP_1 and the problems P_2 and SP_2. That is, if the iREX protocol produces a flow $\boldsymbol{f}$, and the optimal flow for problem P_1 is denoted by $\boldsymbol{f}^*$, the efficiency loss with respect to P_1 is given as:

$$\text{Efficiency loss} = \frac{\Phi(\boldsymbol{f})-\Phi(\boldsymbol{f}^*)}{\Phi(\boldsymbol{f}^*)} \times 100.$$

We show the numerical results in reference to the multiple and single path solutions. Each graph in this section has two curves, which show the efficiency loss with respect to a solution that uses multiple paths depicted by (Multipath), and the efficiency loss with respect to a solution that only uses a single path depicted by (Singlepath). The simple average of the difference between the two curves is also included. While the single path routing describes the iREX protocol, the multipath solution is an absolute reference bound for all possible methods (including future multipath iREX protocol improvements).

Simulation Results

Figures 3.12, 3.13, and 3.14 show the efficiency loss of the iREX protocol compared to optimal solutions of problems P_1 and SP_1 using *linear*, *squared*, and *random* price functions under varying traffic load. For nominal to high traffic loads of 50% or more, the worst case (*random*) efficiency loss to the single path SP_1 metric is about 16%, and the "best" worst case among the three sub configurations is about 12% (*squared*). Price functions determine the speed of a domain's response to increasing load situations. The "faster" squared price function allows for faster use of alternative paths, thereby making the squared

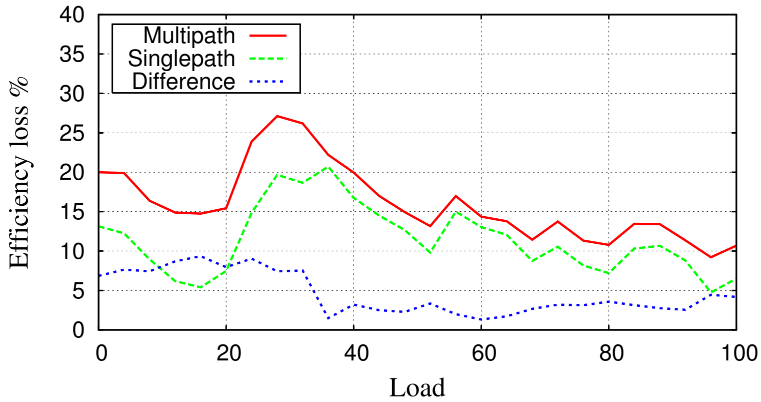

Figure 3.12: Efficiency of the iREX protocol for linear price functions with respect to offline optimum of type P_1 and SP_1.

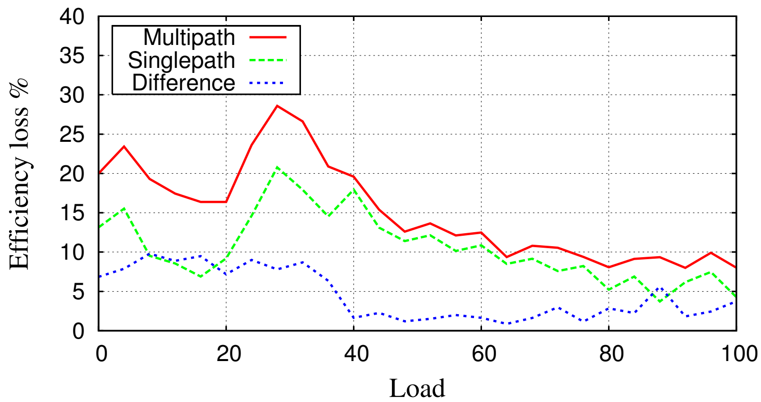

Figure 3.13: Efficiency of the iREX protocol for squared price functions with respect to offline optimum of type P_1 and SP_1.

sub configuration perform better. To further expose this behavior, we refer to the squared (Figure 3.13) sub configuration's smaller distance to the optimal solution in comparison to the *linear* (Fig. 3.12) sub configuration's more pronounced efficiency loss peaks at traffic loads 36%, 60%, 72% and 88%. The *random* (Figure 3.14) sub configuration, which represents the most realistic scenario, performed worse than the domains in the uniform price function sub configurations. This may be caused by the diversity of price functions. We note, however, that the worst efficiency loss difference between the *random* and the best (*squared*) sub configuration is only about 5%. We also observe that the efficiency loss with respect to the single path metric SP_1 is consistently and recognizably lower than the efficiency loss with respect to the multi path metric P_1 with the difference averaging between 4.52% to 4.61%. In all cases, efficiency loss decreases with increased traffic load. The reason is that as traffic load increases, the search space for "good" paths decreases. Figures 3.15, 3.16, and 3.17 show the efficiency loss to P_2 and SP_2 for iREX using the linear, squared, and random price functions respectively under varying traffic load.

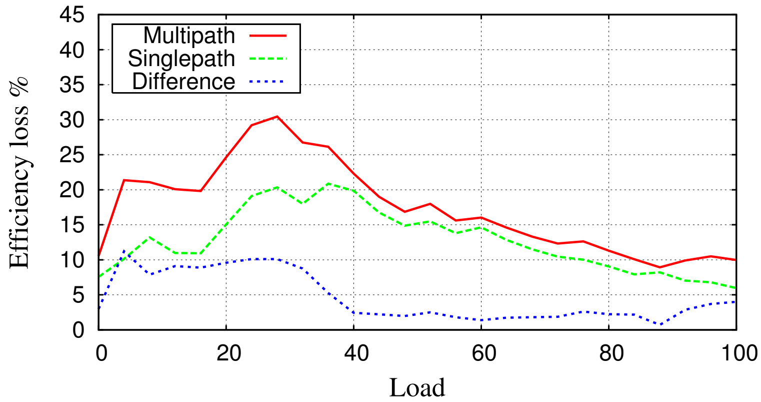

Figure 3.14: Efficiency of the iREX protocol for random price functions with respect to offline optimum of type P_1 and SP_1.

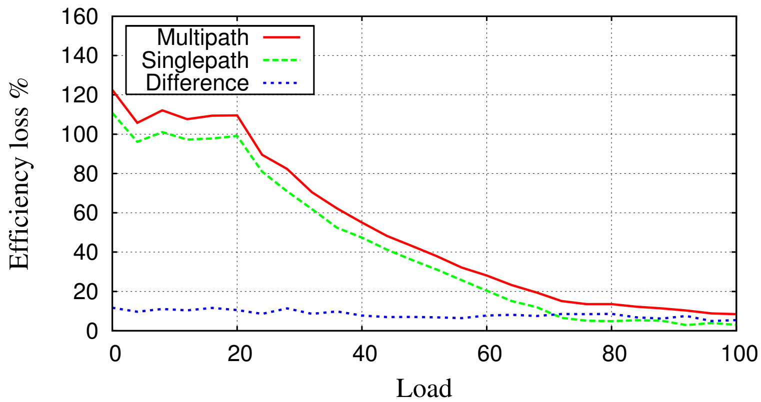

Figure 3.15: Efficiency of the iREX protocol for linear price functions with respect to offline optimum of type P_2 and SP_2.

We again observe that the efficiency loss with respect to the single path metric SP_2 is consistently and recognizably lower than the efficiency loss with respect to the multi path metric P_2 with the difference between the single SP_2 and the multi P_2 path averaging between 8.38% to 8.42%. And again in all cases, efficiency loss decreases with increased traffic load. The differences in the sub configurations are small due to the nature of the metric.

Comparison with Current SLA Method

In contrast to the the iREX protocol, the SLA method exhibits an efficiency loss of 150% which increased to a maximum of 340% in the same traffic load range as seen in Fig. 3.18. The constant increase in efficiency loss is due to the static nature of this method.

The SLA method stays at about 70% efficiency loss across the same traffic load ranges as seen in Fig. 3.19. This efficiency loss does not increase because usage on the most congested arc has reached maximum capacity.

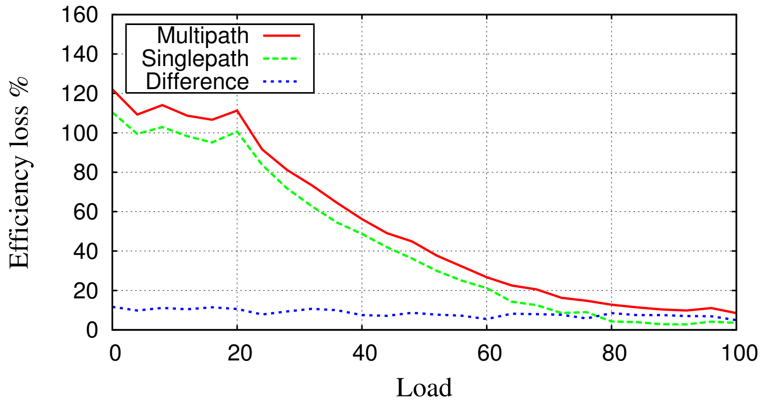

Figure 3.16: Efficiency of the iREX protocol for squared price functions with respect to offline optimum of type P_2 and SP_2.

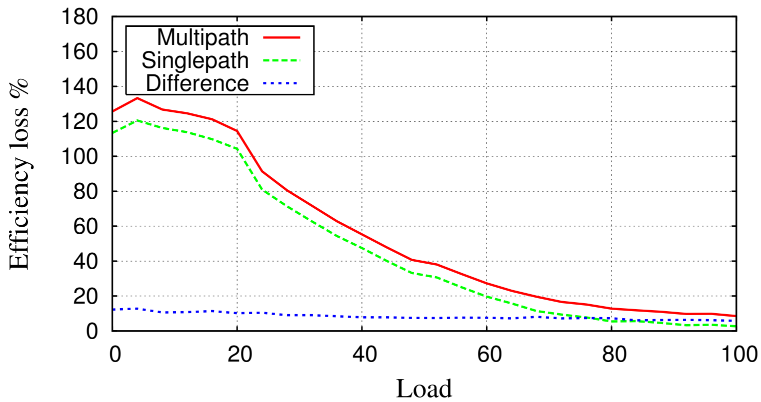

Figure 3.17: Efficiency of the iREX protocol for random price functions with respect to offline optimum of type P_2 and SP_2.

3.7 Discussion and Open Problems

We see the framework introduced in this chapter as a first step towards modeling of real world online multicommodity routing problems. In practice, however, there are many more additional requirements. For instance, routings have to consider capacities, which we ignored in our theoretical approach. With capacities, however, one can easily construct examples in which any online algorithm does not even produce a feasible solution. Further requirements in practice include path length restrictions and survivability issues. Another important point is that in practice routings are only valid until a given time, after which they disappear. This has effects on the cost for future routings. We plan to study this problem in the future. It is also an open issue, whether the competitiveness bound in Theorem 3.9 and Theorem 3.24 are tight, and whether there exists a competitive online algorithm for the unsplittable variant of the OnlineMCRP.

As the last section suggests, for realistic network and traffic instances, the proposed online algorithms are expected to perform better than the provable

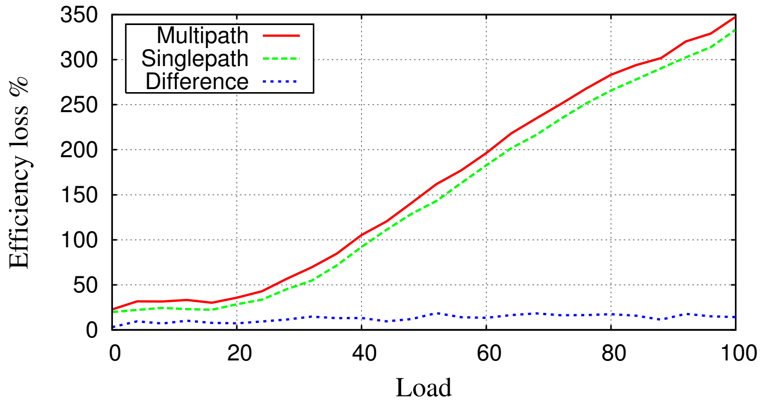

Fig. 3.18: SLA Efficiency loss to to offline optimum of type P_1 and SP_1

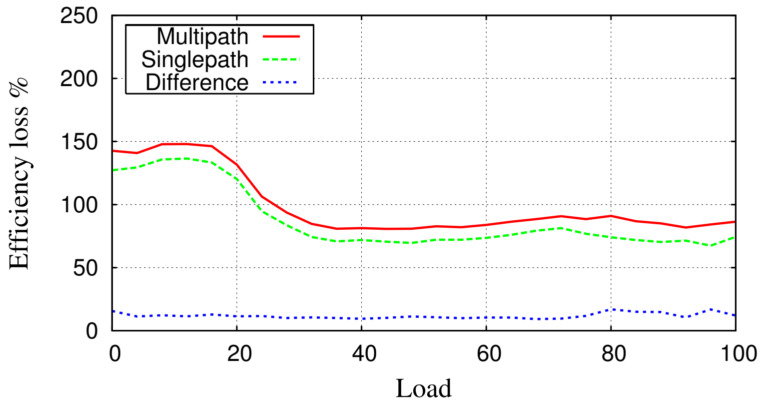

Fig. 3.19: SLA Efficiency loss to offline optimum of type P_2 and SP_2

bounds.

Chapter 4

Network Games

In the last chapter, we presented an online routing problem in which demands in a network have to be routed consecutively. In this chapter, we investigate selfish routing problems or network games. In a network game, players route demand in a network with minimum cost. Congestion on an arc is modeled by a nondecreasing latency function. Such functions map the total flow on an arc to the time needed to traverse this arc. In practical applications, the most frequently used functions are polynomials whose degrees and coefficients are determined from real-world data through statistical evaluation methods, see Patrikkson [73] and Branston [17]. The cost on an arc is defined as latency times flow on that arc. The total cost of routing flow is defined by the sum over all arc costs.

Rosenthal [79] introduced the *atomic unsplittable* model, where players have to route their demands along a single path. He showed that a Nash equilibrium in pure strategies exists when all players control the same amount of flow. Conversely, a Nash equilibrium need not exist when players control different amounts of flow. Milchtaich [67] studied the *nonatomic model*, where a large number of players is assumed, where each player only controls an infinitesimal part of the entire flow. He showed that this variant arises as the limit of a sequence of atomic unsplittable network games, where the number of players goes to infinity. The *atomic splittable* model, where some players may control a significant part of the entire demand, was first considered in the transportation literature, see Catoni and Pallatino [19]. Subsequent work in this area can be found in Orda, Rom, and Shimkin [72] and Roughgarden and Tardos [84]. All these models can be seen as special cases of general non-cooperative congestion games. In non-cooperative games, players select strategies that are subsets of resources, and the utility of a player only depends on the number of players choosing the same or some overlapping strategy, see Rosenthal [79].

In this chapter, we focus on *nonatomic* and *atomic splittable* network games, which we call nonatomic and atomic network game, respectively. Recall that a system optimum is a multicommodity flow with minimum total cost, while a Nash equilibrium amounts to a flow, where no player can improve by switching flow to another path under the prevailing conditions.

The most prominent application of nonatomic network games is the road traffic network in which travelers, usually drivers of vehicles, choose routes from their origins to their destinations. Of particular interest are traffic equilibrium models to describe and predict the arc flows and travel times resulting from an outcome of selfish route selection. The behavioral assumption for nonatomic travelers, known as the Wardrop principle, postulates that the travel times along the used routes for a given source-destination pair are equal to or less than those on unused routes. It can be shown that such "user-optimized" flows are actually Nash equilibria for the corresponding nonatomic network game.

In atomic network games, some players may control a significant part of the entire demand. Aggregating, controlling, and coordinating demand applies to many real-world examples. For instance, route guidance systems are becoming increasingly popular for car drivers. They enter their current position determined via the *Global Positioning System* (GPS) at the beginning of a trip. Then, a central computer calculates an "optimal" route for this trip based on digital maps, and based on available knowledge of traffic congestion on the streets. Since a route guidance operator controls the aggregate traffic of its customers, such an operator is an atomic player in game theoretic terminology.

Logistic and freight companies use trucks, trains or ships to carry goods from source to destination points. These vehicles have to traverse parts of a network that is also shared by other competitors and civil traffic. Some companies may control many such vehicles which makes these companies control a significant part of the overall traffic. Furthermore, the market share of a single player may even increase if freight companies subcontract services from a single logistic company.

We study the competition of atomic players using non-cooperative game theory. Note that the case of a nonatomic player emerges as a special case in which infinitely many atomic players are allowed each of them controlling a negligible amount of flow. In non-cooperative game theory, we rely on the classical equilibrium concept of Nash [71] to analyze an atomic network game. At Nash equilibrium, no player can reduce its cost by switching flow to another path provided all other players keep their routing fixed. In contrast to the nonatomic case, a Nash equilibrium in the atomic case does not necessarily coincide with a Wardrop equilibrium [87]. A trivial example for this is to consider a single atomic player. The Nash equilibrium in this case is equal to the system optimum which does not always hold for a Wardrop equilibrium. The Pigou instance (see Fig. 5.6 for a graphical illustration) is a prominent example for this possibility [75].

4.1 Related Work

In the last years, there has been an exciting development in algorithmic game theory, aiming at quantifying the efficiency loss of Nash equilibria (user equilibria) in non-cooperative games. The fact that there exists an efficiency loss of the user equilibrium compared to a system optimum is well known in the transportation literature, see Braess [16] and Dubey [28]. A first result for exactly quantifying the price of anarchy was given by Papadimitriou and Koutsoupias [62] in the context of a load balancing game in communication networks. Roughgarden and Tardos [84] were able to bound the price of anarchy in nonatomic network games. In particular, Roughgarden and Tardos [84] proved for a set of separable affine cost functions a bound of $\frac{4}{3}$ on the price of anarchy. A series of several other papers analyzed the price of anarchy for more general cost functions and model features; see for example Correa, Schulz, and Stier-Moses [24, 25], Czumaj and Vöcking [26], Perakis [74], and Roughgarden [80].

By introducing the so called anarchy value $\alpha(\mathcal{L})$, Roughgarden [81] proved the first tight bounds on the price of anarchy for nonatomic network games and general nondecreasing, continuous and s-convex latency functions.[1] Correa, Schulz, and Stier-Moses [24] introduced the value $\beta(\mathcal{L})$ and only assumed that latency functions have to be continuous and nondecreasing. They proved that if the anarchy value exists, their bound implies all bounds of Roughgarden by using the relation $\alpha(\mathcal{L}) = \big(1 - \beta(\mathcal{L})\big)^{-1}$.

Even though the bounds obtained by Roughgarden are shown to be tight for classes $\mathcal{L}$ that contain constants functions, there exist classes $\mathcal{L}$ of *homogenous* latency functions, i.e. $\ell_a(0) = 0,\ \forall a \in A$, where the anarchy value $\alpha(\mathcal{L})$ and the value $\beta(\mathcal{L})$ do not lead to tight upper bounds. Consider for example monomial latency functions $\mathcal{M}_d := \{a_d\, x^d :\ a_d \geq 0\}$, of arbitrary degree $d \geq 1$. Using a variational inequality characterizing a Nash flow, it can be shown that the price of anarchy is exactly one in this case, that is, the Nash equilibrium is an optimal flow. But neither the anarchy value $\alpha(\mathcal{M}_d)$, nor the parameter $\beta(\mathcal{M}_d)$ gives the correct upper bound as also mentioned by Roughgarden in [81]. In this regard, Dumrauf and Gairing [29] improved bounds on the price of anarchy for classes $\mathcal{M}_{s,d} := \{a_d\, x^d + \cdots + a_s\, x^s :\ a_j \geq 0, j = s \ldots, d\}$, where $s \geq 1$ is the minimum degree and $d \geq s$ is the maximum degree. Their result, however, is tailored to this particular class and does not provide bounds for more general homogenous latency functions.

Using the parameter $\beta(\mathcal{L})$, Correa, Schulz, and Stier-Moses[24] also showed that for capacitated networks, all known bounds on the price of anarchy without capacities are valid, provided a special Nash equilibrium, called the Beckman equilibrium [11], is under consideration. As described by the same authors, for selfish routing problems with capacities several Nash equilibria may exist. Furthermore, they presented instances, where the efficiency loss of a Nash equilibrium is unbounded. Larsson and Patriksson [64, 73] and Marcotte, Nguyen, and Schoebel [65] proposed to include explicit arc capacities as an obvious way to improve the quality of traffic assignment models. Indeed, the widely used link delay formula proposed by the Bureau of Public Roads includes a capacity parameter [18]. A frequently used way to implicitly incorporate capacities is to use the so called volume delay formulas that tend

[1] A function $\ell(x)$ is called s-convex, if the function $\ell(x)\, x$ is convex.

to infinity as the arc flow approaches the arc capacity, see Branston [17] for a discussion. In a related application, the introduction of capacities can be used to derive tolls for the reduction of flows on overloaded links, see Hearn and Ramana [54]. Further work on network tolls as a way to improve the performance of the user equilibrium can be found by Cole, Dodis, and Roughgarden [21, 22]. In particular, they prove that in case users have different valuations of delay, there exists a set of optimal tolls for a single commodity network. Fleischer, Jain, and Mahdian [35] extended this result to general networks. Jahn, Möhring, and Stier-Moses [55] empirically investigate the performance of user equilibria with latency constraints for users. All models mentioned so far, assume static multicommodity flow networks. Only a few results are known for time dependent multicommodity flow problems, see for instance Köhler and Skutella [61] and Hall, Langkau, and Skutella [50].

For atomic network games and unsplittable flow, Roughgarden and Tardos examined the price of anarchy [84]. Awerbuch, Azar, and Epstein [10] and Christodoulou and Koutsoupias [20] studied the price of anarchy in the unsplittable variant for linear latency functions. Aland et al. [3] then proved exact bounds on the price of anarchy for general polynomial latency functions in this case.

The splittable atomic case was first considered by Orda, Rom, and Shimkin in [72]. They prove the existence of Nash equilibria by relying on the classical result about concave games obtained by Rosen in [78]. Further results about the uniqueness of Nash equilibria are presented by Milchtaich [68, 69] and Richman and Shimkin in [77]. Hayrapetyan, Tardos, and, Wexler [53] presented bounds on the price of anarchy for splittable flows in special network topologies. Fotakis, Kontogiannis, and Spirakis [40] studied algorithmic issues in the same setting. Roughgarden [83] introduced the value $\alpha^K(\mathcal{L})$ and proved that the cost of a flow at Nash equilibrium is upper bounded by this value for general networks. Correa, Schulz, and Stier-Moses [25] proposed the value $\beta^K(\mathcal{L})$ and showed that for classes $\mathcal{L}$ in which $\alpha^K(\mathcal{L})$ exists, the relation $\alpha^K(\mathcal{L}) = (1 - \beta^K(\mathcal{L}))^{-1}$ is valid. Both groups claimed that the price of anarchy in the atomic network game does not exceed that of the corresponding nonatomic one. This turned out to be wrong as discovered by Cominetti, Correa, and Stier-Moses in [23]. Based on the work of Catoni and Pallotino [19], they presented an example, where the price of anarchy in a network game with atomic players is larger than that of the corresponding nonatomic game. Moreover, they showed that by aggregating and controlling demand the cost for this aggregate may even increase compared to the game without aggregation. Such a counter-intuitive phenomenon can also arise from the perspective of single individuals: players outside the cartel may face lower cost compared to the situation, in which this player competes with the individuals instead of the cartel.

Despite the possible increased efficiency loss of equilibria in atomic network games compared to the nonatomic counterpart, Cominetti, Correa, and Stier-Moses showed that the price of anarchy can be bounded for special latency functions in this case. In particular, they proved upper bounds on the price of anarchy for affine linear, squared, and cubic latency functions of $1.5, 2.56$, and 7.83, respectively. They obtain these results by correctly analyzing the value $\beta^K(\mathcal{L})$. For general polynomials, however, their approach fails to generate upper bounds.

4.2 Contributions and Chapter Outline

We study network games with nonatomic and atomic players and spittable flow. For network games with nonatomic players and for a class of functions $\mathcal{L}$, we introduce the parameter $\omega(\mathcal{L}, \lambda)$ that generalizes the anarchy value $\alpha(\mathcal{L})$ and the value $\beta(\mathcal{L})$. Using this value, we reprove the existing tight bounds on the price of anarchy and present a novel proof for monomial latency functions showing that the price of anarchy is one in this case.

For network games with atomic players, we introduce the value $\omega(\mathcal{L}; K, \lambda)$ that generalizes the previous parameters $\alpha^K(\mathcal{L})$ and $\beta^K(\mathcal{L})$, which were proposed by Roughgarden [83] and Correa, Schulz, and Stier-Moses [25], and Cominetti , Correa, and Stier-Moses [23], respectively. For classes $\mathcal{L}$ for which $\alpha^K(\mathcal{L})$ and $\beta^K(\mathcal{L})$ exists, the relation $\omega(\mathcal{L}; K, 1) = \beta^K(\mathcal{L})$ is fulfilled. With a proper choice of λ, however, we are able to improve all existing bounds, except for the case of affine linear latency functions. In the case of affine linear latency functions, we show that indeed $\lambda = 1$ is the best choice in our approach.

We start in Section 4.3 by introducing the basic traffic model for nonatomic network games. In Section 4.3.4, we present a generalized method for bounding the price of anarchy that extends previous work of Roughgarden [81], and Correa, Schulz, and Stier-Moses [24].

In Section 4.4 we study network games with atomic players. The response strategy of an atomic player can be described by an associated optimization problem, where the objective is to minimize the individual cost. Under mild assumptions on feasible latency functions this problem is a convex problem. This type of problem can be solved within arbitrary precision in polynomial time. In Section 4.4.2, we present a technique to bound the price of anarchy for atomic network games. This technique also generalizes concepts from Roughgarden [83], Correa, Schulz, and Stier-Moses [25], and Cominetti , Correa, and Stier-Moses [23]. Equipped with this technique, we present bounds on the price of anarchy for polynomial latency functions with nonnegative coefficients that improve all previous results, except for affine linear latency functions. For an overview of these results see Table 4.3.

4.3 Nonatomic Network Games

The traffic model for selfish network games is similar to the multicommodity flow problem presented in Chapter 3. The main difference is that commodities are not released online but are considered at the same time in parallel. Furthermore, the considered cost function is different. In Chapter 3, the routing cost on an arc a is defined as the integral over the aggregate flow on a with respect to a nondecreasing price function $p_a(\cdot)$. There, the costs represent monetary costs for routing the commodities. In this chapter, arcs are equipped with latency functions $\ell_a(\cdot)$ and we define the routing cost on an arc as the latency multiplied with the total flow on that arc.

In the following, we use the same notation as in Chapter 3. An instance of the nonatomic network game is given by the triple (D, d, ℓ), where $D = (V, A)$ represents a directed graph. Furthermore, we are given a set of commodities $[K] := \{1, \ldots, K\}$, where each commodity $k \in [K]$ has a demand $d_k > 0$ that has to be routed from a source $s_k \in V$ to a destination $t_k \in V$. The latency or delay on an arc $a \in A$ is given by a nondecreasing continuous and separable latency functions $\ell_a : \mathbb{R}_+ \to \mathbb{R}_+$. A latency function $\ell_a(\cdot)$ is called separable if the latency of a feasible flow $\boldsymbol{f}$ on arc a depends on the total flow f_a on a only.

In many cases, it is convenient to assume that the expression $\ell_a(z)\, z$ is a convex function, or s-convex, see Bergendorf, Hearn, and Ramana [13]. Whenever s-convexity is required, we indicate this.

In network games with *nonatomic* players, it is assumed that the flow $\boldsymbol{f}^k$ of commodity k is carried by a large number of agents each controlling an infinitesimal fraction of the entire demand d_k. Thus, the route choice of a single agent does not affect the travel time of others. The travel time for the flow $\boldsymbol{f}$ on the path P is given by

$$\ell_P(\boldsymbol{f}) := \sum_{a \in P} \ell_a(f_a).$$

We define the total travel time as the sum of travel times on arcs of the network:

Definition 4.1 (Total Cost)
The total cost for a flow $\boldsymbol{f}$ is given by:

$$C(\boldsymbol{f}) := \sum_{P \in \mathcal{P}} \ell_P(\boldsymbol{f})\, f_P. \tag{4.1}$$

The total cost can also be represented by the sum of arc costs:

$$\sum_{P \in \mathcal{P}} \ell_P(\boldsymbol{f})\, f_P = \sum_{P \in \mathcal{P}} \Big(\sum_{a \in P} \ell_a(f_a) \Big) f_P = \sum_{a \in A} \Big(\sum_{\substack{P \in \mathcal{P} \\ a \in P}} f_P \Big) \ell_a(f_a) = \sum_{a \in A} \ell_a(f_a)\, f_a.$$

The first equation holds since latency functions are assumed to be separable. The second and third equation follows from changing the summation order. Hence, we have

$$C(\boldsymbol{f}) = \sum_{a \in A} \ell_a(f_a)\, f_a.$$

We used that the path decomposition of a flow defines a unique decomposition into arc flows. Conversely, an arc decomposition of a flow may be represented by several path decompositions.

4.3.1 The Nash Equilibrium for Nonatomic Players

The basic assumption in this thesis is that users (players) act selfishly. This means that players are solely interested in maximizing their own individual utility rather than caring about social welfare.

To analyze the outcome of such individual behavior, one usually tries to analyze an equilibrium situation: a stable point from which no player deviates unilaterally. A flow for the nonatomic network game is a *Wardrop* equilibrium, if for every source-destination pair the latency for the used routes are equal to or less than those on unused routes. This concept was introduced by Wardrop (1952) in his first principle [87].

Definition 4.2 (Wardrop Equilibrium [87])
A feasible flow $\boldsymbol{f}$ is a *Wardrop equilibrium* if

$$l_P(\boldsymbol{f}) \leq l_Q(\boldsymbol{f}), \text{for all } k \in [K] \text{ and all paths } P, Q \in \mathcal{P}_k, \text{ such that } f_P > 0. \quad (4.2)$$

A similar concept for general non-cooperative games was proposed at the same time by Nash [71]. A flow (strategy distribution) is at Nash equilibrium if no player has an incentive to unilaterally deviate from the current strategy. This triggers the following definition in the context of nonatomic network games, see also Roughgarden [82].

Definition 4.3 (Nash Equilibrium [71])
A feasible flow $\boldsymbol{f}$ is at *Nash equilibrium*, if routing of a small bundle of flow along another path does not strictly decrease the travel time along this path. Formally, we define for every $k \in [K]$, and every two paths $P, Q \in \mathcal{P}_k$, such that $f_P > 0$, and $0 \leq \epsilon \leq f_P$, a flow $\boldsymbol{f}^\epsilon$ by

$$f_P^\epsilon = \begin{cases} f_R - \epsilon & \text{if } R = P \\ f_R + \epsilon & \text{if } R = Q \\ f_R & \text{otherwise,} \end{cases} \quad (4.3)$$

where $P \in \mathcal{P}$. Then, a feasible flow $\boldsymbol{f}$ is a *Nash equilibrium* if $l_P(\boldsymbol{f}) \leq l_Q(\boldsymbol{f}^\epsilon)$ for all $\epsilon \in [0, f_P]$.

It can be shown that a Nash flow $\boldsymbol{f}$ solves the following convex optimization problem, see for example Roughgarden and Tardos [84].

$$\begin{aligned} \min \quad & \sum_{a \in A} \int_0^{f_a} \ell_a(z)\, dz \\ \text{s.t.} \quad & \sum_{a \in \delta^+(v)} f_a^k - \sum_{a \in \delta^-(v)} f_a^k = \gamma(v) && \forall\, v \in V,\ k \in [K] \\ & f_a^k \geq 0 && \forall\, a \in A,\ k \in [K], \end{aligned} \quad (4.4)$$

where $\gamma(v)$ is defined as in (3.1).

Note that convexity already follows from the assumption that latency functions are nondecreasing.

Remark 4.4. The above characterization of a flow at Nash equilibrium implies that every instance (D, d, ℓ) admits a Nash equilibrium. To see this, consider the convex program (4.4). This problem has a continuous objective and a bounded and closed feasible region. Hence, the existence of an optimal solution is assured. Furthermore, the convexity of the objective implies that the optimal value is unique. Thus, every flow at Nash equilibrium has the same cost.

The following conditions are necessary and sufficient to characterize a Nash equilibrium for a nonatomic network game.

Lemma 4.5. *A feasible flow $\boldsymbol{f}$ is at Nash equilibrium if and only if it satisfies:*

$$\sum_{a \in A} \ell_a(f_a)\,(f_a - x_a) \leq 0 \text{ for all feasible flows } \boldsymbol{x}. \tag{4.5}$$

The proof is based on the first order optimality conditions and the convexity of the cost function in (4.4), see Dafermos and Sparrow [27].

4.3.2 The System Optimum

A central network manager would try to find a routing assignment $\boldsymbol{f}$ that minimizes the total travel time for all commodities. Formally, such a flow solves the problem:

$$\begin{aligned} \min \quad & \sum_{a \in A} \ell_a(f_a) f_a & & \\ \text{s.t.} \quad & \sum_{a \in \delta^+(v)} f_a^k - \sum_{a \in \delta^-(v)} f_a^k = \gamma(v) & & \forall\, v \in V,\ k \in [K] \\ & f_a^k \geq 0 & & \forall\, a \in A,\ k \in [K], \end{aligned} \tag{4.6}$$

where $\gamma(v)$ is defined as in (3.1).

If the latency functions are s-convex, this problem can be efficiently solved within arbitrary precision in polynomial time using the ellipsoid method, see Grötschel, Lovász, and Schrijver [48]. For latency functions that are s-convex, the following conditions are necessary and sufficient to characterize a system optimal flow.

Lemma 4.6. *Let the latency functions be s-convex. A feasible flow f solves* (4.6) *if and only if it satisfies:*

$$\sum_{a \in A} \Big(\ell_a(f_a) + \ell_a'(f_a)\, f_a\Big)\,(f_a - x_a) \leq 0 \text{ for all feasible flows } x. \tag{4.7}$$

The proof is based on the first order optimality conditions and the convexity of the objective function, see Dafermos and Sparrow [27]. Note that the only difference to the characterization of a flow at Nash equilibrium is the

Table 4.1: Price of Anarchy for different polynomial latency functions. All coefficients a_i are assumed to be nonnegative.

Allowable cost functions $\mathcal{L}$	Example	Price of Anarchy $\alpha(\mathcal{L})$
affine linear functions	$a_1 x + a_0$	1.334
quadratic functions	$\sum_{i=0}^{2} a_i x^i$	1.626
cubic functions	$\sum_{i=0}^{3} a_i x^i$	1.896
polynomials of degree 4	$\sum_{i=0}^{4} a_i x^i$	2.151
polynomials of degree d	$\sum_{i=0}^{d} a_i x^i$	$\frac{(d+1)\,(d+1)^{(1/d)}}{(d+1)\,(d+1)^{(1/d)}-d}$

term $\ell'_a(f_a)\, f_a$ arising from the derivative of $\ell_a(f_a)\, f_a$. In this regard, Dafermos and Sparrow [27] proved that for latency functions satisfying

$$\ell_a(f_a) + \ell'_a(f_a)\, f_a = \kappa\, \ell_a(f_a),$$

for some nonnegative number κ, the cost of a flow at Nash equilibrium is equal to the system optimal cost.

4.3.3 Price of Anarchy

A natural question that arises in the context of a Nash equilibrium is: How efficient is a Nash equilibrium compared to the system optimum? For answering this question for network games, we need to analyze the worst case ratio between the cost of a flow at Nash equilibrium and that of a system optimal flow, see Papadimitriou and Koutsoupias [62] and Roughgarden and Tardos [84].

Definition 4.7 (Price of Anarchy)
Let (D, d, ℓ) be an instance of a nonatomic routing game. The *price of anarchy* of the instance (D, d, ℓ) is denoted by $\rho(D, d, \ell)$ and defined as:

$$\rho(D, d, \ell) = \frac{C(\boldsymbol{f})}{C(\boldsymbol{f}^*)}. \tag{4.8}$$

If $\mathcal{I}$ is the set of all instances, then the price of anarchy of $\mathcal{I}$ is:

$$\rho(\mathcal{I}) = \sup_{(D,d,\ell)\in\mathcal{I}} \rho(D, d, \ell).$$

The first tight bounds for general polynomial latency functions were presented by Roughgarden and Tardos [84] and Roughgarden [81]. For a given class $\mathcal{L}$ of latency functions that are continuous, nondecreasing, differentiable, and s-convex, Roughgarden defined the so called *anarchy value* $\alpha(\mathcal{L})$ as

$$\alpha(\ell_a) := \sup_{x_a, f_a \geq 0} \frac{\ell_a(f_a)\, f_a}{\ell_a(x_a)\, x_a + (f_a - x_a)\, \ell_a(f_a)} \tag{4.9}$$
$$\alpha(\mathcal{L}) := \sup_{\ell_a \in \mathcal{L}} \alpha(\ell_a).$$

Equipped with this definition we restate his result.

Theorem 4.8 (Roughgarden [81]). *Let $\mathcal{L}$ be a set of latency functions with anarchy value $\alpha(\mathcal{L})$, and (D, d, ℓ) an instance with latency functions in $\mathcal{L}$. Then, the price of anarchy for this instance is at most $\alpha(\mathcal{L})$.*

In Table 4.1 the exact price of anarchy for polynomial latency functions is shown. Correa, Schulz, and Stier-Moses [24] defined a similar value $\beta(\mathcal{L})$.

$$\beta(\ell_a) := \sup_{x_a, f_a \geq 0} \frac{\big(\ell_a(f_a) - \ell_a(x_a)\big)\, x_a}{\ell_a(f_a)\, f_a} \tag{4.10}$$
$$\beta(\mathcal{L}) := \sup_{\ell_a \in \mathcal{L}} \beta(\ell_a).$$

For classes $\mathcal{L}$ for which $\alpha(\mathcal{L})$ exists, these two values are related by the equation $\alpha(\mathcal{L}) = \big(1 - \beta(\mathcal{L})\big)^{-1}$. In the next section, we present a detailed discussion about these values.

4.3.4 Bounding the Price of Anarchy

In the following, we derive upper bounds on the price of anarchy by introducing the parameter $\omega(\mathcal{L}, \lambda)$ that generalizes the anarchy value $\alpha(\mathcal{L})$ and the value $\beta(\mathcal{L})$. With this value, we reprove the existing tight bounds on the price of anarchy and present a novel proof for monomial latency functions showing that the price of anarchy is one in this case.

For every arc a, latency function ℓ_a, and nonnegative number λ, we define the following nonnegative value:

$$\omega(\ell_a; \lambda) := \sup_{f_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a}{\ell_a(f_a)\, f_a}. \tag{4.11}$$

We assume by convention $0/0 = 1$. For a given class $\mathcal{L}$ of nondecreasing latency functions, we further define

$$\omega(\mathcal{L}; \lambda) := \sup_{\ell_a \in \mathcal{L}} \omega(\ell_a; \lambda).$$

See Figure 4.1 for a graphical illustration of this value. Before we state the main theorem, we define the following:

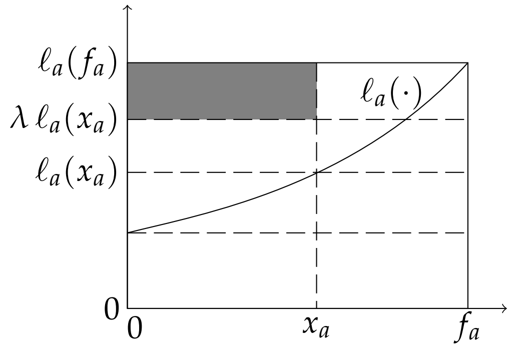

Figure 4.1: Illustration of the value $\omega(\ell_a;\lambda)$ in equation (4.11) with $1 < \lambda < \frac{\ell_a(f_a)}{\ell_a(x_a)}$. The gray-shaded area corresponds to the value $\omega(\ell_a;\lambda)$.

Definition 4.9
We define the set of feasible $\lambda \geq 0$ as

$$\Lambda(\mathcal{L}) := \{\lambda \in \mathbb{R}^+ \mid \big(1 - \omega(\mathcal{L};\lambda)\big) > 0\}.$$

Theorem 4.10. *For latency functions in $\mathcal{L}$, the price of anarchy for the nonatomic network game is at most*

$$\inf_{\lambda \in \Lambda(\mathcal{L})} \left[\lambda\,(1 - \omega(\mathcal{L};\lambda))^{-1}\right].$$

Proof. Let $\boldsymbol{f}$ be a flow in Nash equilibrium, and let $\boldsymbol{x}$ be any feasible flow. Then, we have

$$\begin{aligned} C(\boldsymbol{f}) &= \sum_{a \in A} \ell_a(f_a)\, f_a \\ &\leq \sum_{a \in A} \ell_a(f_a)\, x_a && (4.12) \\ &= \sum_{a \in A} \ell_a(f_a)\, x_a + \lambda\, \ell_a(x_a)\, x_a - \lambda\, \ell_a(x_a)\, x_a \\ &\leq \lambda\, C(\boldsymbol{x}) + \omega(\mathcal{L};\lambda)\, C(\boldsymbol{f}). && (4.13) \end{aligned}$$

Here, (4.12) follows from the variational inequality stated in Lemma 4.5. The last inequality (4.13) follows from the definition of $\omega(\mathcal{L};\lambda)$. Taking $\boldsymbol{x}$ as the optimal offline solution and since $\lambda \in \Lambda(\mathcal{L})$, the claim is proven. □

The last step in the proof justifies the rather cryptic definition of $\Lambda(\mathcal{L})$. Note that the infimum in Theorem 4.10 can be infinite and the set $\Lambda(\mathcal{L})$ can be empty.

4.3.5 Comparison with Previous Results

Let $\mathcal{L}$ be a class of latency functions. In the following we relate the value $\omega(\mathcal{L};\lambda)$ to the anarchy value $\alpha(\mathcal{L})$ introduced by Roughgarden in [81] and to the parameter $\beta(\mathcal{L})$ introduced by Correa, Schulz, and Stier-Moses in [24].

In the original setting of Roughgarden, he assumed for the definition of $\alpha(\mathcal{L})$ that the class $\mathcal{L}$ consists of continuous, nondecreasing, differentiable, and s-convex functions. For this class he showed that the anarchy value $\alpha(\mathcal{L})$ is tight by presenting matching lower bounds. Correa, Schulz, and Stier-Moses relaxed the assumptions of differentiability and s-convexity by only assuming that latency functions have to be continuous and nondecreasing (in fact only lower semi continuity is required as shown in Correa, Schulz, and Stier-Moses in [24]). For classes $\mathcal{L}$ in which $\beta(\mathcal{L})$ exists, they proved that their bound implies all bounds of Roughgarden by using $\alpha(\mathcal{L}) = \big(1 - \beta(\mathcal{L})\big)^{-1}$. Moreover using $\beta(\mathcal{L})$, they extended the analysis to capacitated networks.

Our definition of $\omega(\mathcal{L};\lambda)$ is equal to $\beta(\mathcal{L})$ if we set $\lambda = 1$. In this regard, the parameter $\omega(\mathcal{L};\lambda)$ is as general as $\beta(\mathcal{L})$ in that we only require continuity and monotonicity for feasible latency functions. We will show, however, that for a wide class of latency functions the assumption $\lambda = 1$ leads to tight bounds, if the class of allowable latency functions contains the constant functions. A prominent example highlighting this issue are the class of monomial latency functions $\mathcal{M}_d = \{\ell(x) = a_d\, x^d : a_d \in \mathbb{R}^+\}$ of arbitrary degree $d \in \mathbb{N}$. Using the variational inequality stated in Lemma 4.5 it can be shown that the price of anarchy is exactly one, see Dafermos and Sparrow [27]. But neither the anarchy value $\alpha(\mathcal{M}_d)$, nor the parameter $\beta(\mathcal{M}_d)$ gives the correct upper bound as also mentioned by Roughgarden in [81]. With our approach, we obtain the correct bounds for monomial latency functions as shown in Section 4.3.7.

In the following, we consider the class $\mathcal{L}_d$ of polynomials with nonnegative coefficients and degree at most $d \in \mathbb{N}$:

$$\mathcal{L}_d := \{a_d\, x^d + \cdots + a_1\, x + a_0 \;:\; a_s \geq 0, s = 0, \ldots, d\}.$$

Furthermore, we analyze latency functions that are represented by monomials with nonnegative coefficients:

$$\mathcal{M}_d = \{\ell(x) = a_d\, x^d : a_d \in \mathbb{R}^+\}, d \geq 1.$$

4.3.6 Affine Linear and Linear Latency Functions

To demonstrate the potential of Theorem 4.10, we reprove the bound on the price of anarchy for latency functions in $\mathcal{L}_1$. Thereby, we explicitly show that $\lambda = 1$ is an optimal choice for affine linear latency functions. Note that for $\lambda = 1$ we have $\alpha(\mathcal{L}) = \big(1 - \omega(\mathcal{L};1)\big)^{-1}$ and $\beta(\mathcal{L}) = \omega(\mathcal{L};1)$.

Theorem 4.11 (Roughgarden and Tardos [84]). *Let f be a Nash equilibrium of a nonatomic network game with latency functions in $\mathcal{L}_1$. Then,*

$$C(\boldsymbol{f}) \leq \frac{4}{3}\, C(\boldsymbol{x}),$$

for any feasible flow x.

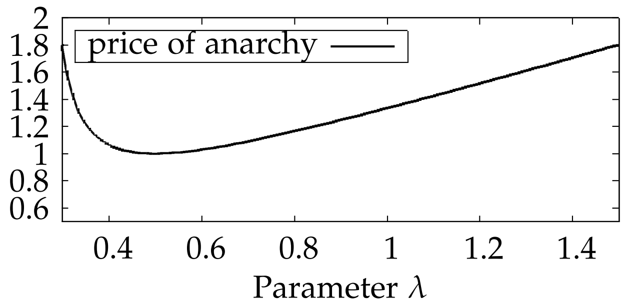

Figure 4.2: The price of anarchy for linear latency functions as a function of the parameter λ.

Proof. We present a proof along the lines of Theorem 4.10. We assume latency functions of the form $\ell_a(z) = q_a\, z + r_a,\ q_a \geq 0,\ r_a \geq 0$. By the definition of $\omega(\mathcal{L};\lambda)$, we have:

$$\begin{aligned}\omega(\ell_a;\lambda) &= \sup_{f_a,x_a\geq 0} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)\,x_a}{\ell_a(f_a)\,f_a}\\ &= \sup_{f_a,x_a\geq 0} \frac{\big(q_a\,f_a - \lambda\,q_a\,x_a\big)\,x_a + \big(r_a - \lambda\,r_a\big)\,x_a}{q_a\,(f_a)^2 + r_a\,f_a}.\end{aligned}$$

The term $\big(r_a - \lambda\,r_a\big)\,x_a$ inside the supremum leads to the condition $\lambda \geq 1$, since otherwise we can set $q_a = 0$ and let x_a tend to infinity to make the supremum unbounded. Hence,

$$\Lambda(\mathcal{L}) = \{\lambda \in \mathbb{R} \mid \lambda \geq 1\}.$$

For $\lambda \geq 1$ we can bound the supremum as follows.

$$\omega(\ell_a;\lambda) \leq \frac{\big(f_a - \lambda\,x_a\big)\,x_a}{(f_a)^2} \leq \frac{1}{4\,\lambda}.$$

Applying Theorem 4.10 yields

$$C(\boldsymbol{f}) \leq \inf_{\lambda\geq 1} \frac{\lambda}{1 - \frac{1}{4\,\lambda}}\, C(\boldsymbol{x}) = \inf_{\lambda\geq 1} \frac{4\,\lambda^2}{4\,\lambda - 1}\, C(\boldsymbol{x})$$

Finally, an easy calculation computes the infimum

$$\min_{\lambda\geq 1} \frac{4\,\lambda^2}{4\,\lambda - 1} = \frac{4}{3}. \tag{4.14}$$

The optimal value is $\lambda^* = 1$, which proves the claim. □

It is easy to show that the main restriction $\lambda \geq 1$ in the proof also holds for general latency functions if constant terms are allowed. The proof indicates that there is potential in improving upper bounds on the price of anarchy for latency functions without affine terms, i.e., $r_a = 0$. Figure 4.3.6 shows the function $\frac{4\lambda^2}{4\lambda-1}$ inside the infimum in (4.14). To precisely quantify this potential, we reprove a well known result obtained by Dafermos and Sparrow [27].

Corollary 4.12 (Dafermos and Sparrow [27]). *Let f be a Nash equilibrium of a nonatomic network game with latency functions in $\mathcal{M}_1$. Then,*

$$C(\boldsymbol{f}) \leq C(\boldsymbol{x}),$$

for any feasible flow $\boldsymbol{x}$.

Proof. We start with (4.14) in the preceeding proof. Analyzing the feasible set $\Lambda(\mathcal{L})$, we have that $\left(1 - \frac{1}{4\lambda}\right) > 0$, leading to $\lambda > \frac{1}{4}$. Thus, we have:

$$\inf_{\lambda > \frac{1}{4}} \frac{4\lambda^2}{4\lambda - 1} = 1,$$

where we have used the optimal value $\lambda^* = \frac{1}{2}$. □

4.3.7 Monomial Latency Functions

Proposition 4.13. *Consider latency functions in $\mathcal{M}_d$ and assume $\lambda > 0$. Then, the value $\omega(\ell_a; \lambda)$ is at most*

$$\frac{d}{(d+1)\left((d+1)\lambda\right)^{\frac{1}{d}}}.$$

Proof. By the definition of $\omega(\ell_a; \lambda)$, we have

$$\omega(\ell_a, \lambda) = \sup_{f_a, x_a \geq 0} \frac{\left(\ell_a(f_a) - \lambda\, \ell_a(x_a)\right) x_a}{\ell_a(f_a)\, f_a}. \tag{4.15}$$

Defining $\mu := \begin{cases} \frac{x_a}{f_a}, & \text{for } f_a > 0 \\ 0, & \text{for } f_a = 0, \end{cases}$ we have to solve

$$\max_{0 \leq \mu} \frac{\left(a_d\, f_a^d - \lambda\, a_d\, \mu^d\, f_a^d)\right) \mu\, f_a}{a_d\, f_a^{d+1}} = \max_{0 \leq \mu} \left(\mu - \lambda\, \mu^{d+1}\right). \tag{4.16}$$

Since this is a strictly convex program, the unique global optimum is given by

$$\mu^* = \left(\frac{1}{(d+1)\lambda}\right)^{\frac{1}{d}}.$$

Note that $\mu^* > 0$ for $\lambda > 0$. Inserting the value μ^* into (4.16) yields the claim. □

Theorem 4.14 (Dafermos and Sparrow [27]). *Let $\mathcal{M}_d$ be the class of allowable latency functions. Then, the price of anarchy for the nonatomic congestion game can be bounded by one.*

Proof. By Proposition 4.13 we can bound $\omega(\ell_a;\lambda)$ by

$$\omega(\ell_a;\lambda) \leq \frac{d}{(d+1)\left((d+1)\,\lambda\right)^{\frac{1}{d}}}.$$

By taking $\lambda^* := \frac{1}{d+1}$ we have $\omega(\ell_a;\lambda) \leq \frac{d}{d+1}$ and hence $\lambda^* \in \Lambda(\mathcal{M}_d)$. Then, applying Theorem 4.10 proves the claim. $\square$

Note that most previous proofs for monomial latency functions use the fact that the variational inequality given in Lemma 4.5 coincides with the conditions of the system optimum.

4.4 Atomic Network Games

In atomic network games, players control and coordinate the entire flow of their demand. The routing strategy of an atomic player amounts to solving an optimization problem, where the objective is to minimize the cost of the demand that the atomic player controls. In this regard, the structure of such a problem is similar to the system optimum presented in Section 4.3. A strategy distribution or flow $\boldsymbol{f}$ is at Nash equilibrium, if no player has an incentive to unilaterally change his strategy. It is straight forward to check that the best reply strategy for player k is the optimum of the following optimization problem, see for example Roughgarden and Tardos [84].

$$\begin{aligned}
\min \quad & \sum_{a\in A} \ell_a(f_a)\, f_a^k \\
\text{s.t.} \quad & \sum_{a\in\delta^+(v)} f_a^k - \sum_{a\in\delta^-(v)} f_a^k = \gamma_k(v) && \forall\, v \in V,\ k \in [K] \\
& f_a^k \geq 0 && \forall\, a \in A,\ k \in [K],
\end{aligned} \tag{4.17}$$

where $\gamma_k(v)$ is defined as in (3.1). In order to have a precise characterization of the solution of the above problem we assume that allowable latency functions are s-convex, that is, $\ell_a(z)\,z$ is a convex function for all $a \in A$. Then, the following conditions are necessary and sufficient to characterize a Nash equilibrium for an atomic routing game.

Lemma 4.15. *A feasible flow $\boldsymbol{f}$ is at Nash equilibrium if and only if for every $k \in [K]$ the following inequality is satisfied:*

$$\sum_{a\in A} \left(\ell_a(f_a) + \ell'_a(f_a)\, f_a^k\right)(f_a^k - x_a^k) \leq 0 \textit{ for all feasible flows } \boldsymbol{x}^k. \tag{4.18}$$

The proof is based on the first order optimality conditions and the convexity of $\ell_a(z)\,z$, see Dafermos and Sparrow [27]. Intuitively, the second term in the derivative of the cost function $\ell'_a(f_a)\,f_a^k$ accounts for the ability of player k to coordinate the flow that it controls.

Remark 4.16. The above characterization of a flow at Nash equilibrium implies that every instance (D, d, ℓ) admits a Nash equilibrium. This follows from a classical result of Rosen [78] that requires convexity of the objective. By the same argument the cost of a flow at Nash equilibrium is unique.

4.4.1 Known Upper Bounds on the Price of Anarchy

The price of anarchy in network games with atomic players and splittable flow has been investigated by Rouhghgarden [83], and Correa, Schulz and Stier-Moses [25]. We summarize in the following the main known results in this field. Roughgarden [83] defined for a given class $\mathcal{L}$ of latency functions the following value ($0/0 = 0$ by assumption):

$$\alpha^K(\ell_a) := \sup_{\boldsymbol{f}_a, x_a \geq 0} \frac{\ell_a(f_a)\,f_a}{\ell_a(x)\,x_a + \sum\limits_{k\in[K]} \big(f_a^k - x_a^k\big)\big(\ell_a(f_a) + \ell'_a(f_a) f_a^k\big)}, \tag{4.19}$$

where the constraint $\boldsymbol{f}_a \geq 0$ is defined as

$$f_a^k \geq 0, \text{ for all } k \in [K] \text{ with } \sum_{k\in[K]} f_a^k = f_a. \tag{4.20}$$

Roughgarden proved that $\alpha^K(\mathcal{L}) := \sup\limits_{\ell_a\in\mathcal{L}} \alpha^K(\ell_a)$ is an upper bound on the price of anarchy of atomic network games.

Proposition 4.17 (Roughgarden[83]). *Consider an atomic network game with K players and latency functions in $\mathcal{L}$. Let $\boldsymbol{f}$ be a Nash equilibrium and let $\boldsymbol{x}^*$ be a social optimum. Then,*

$$C(\boldsymbol{f}) \leq \alpha^K(\mathcal{L})\,C(\boldsymbol{x}^*).$$

Proof. Using the definition of $\alpha^K(\mathcal{L})$, it is easy to see that

$$C(\boldsymbol{x}^*) \geq \frac{C(\boldsymbol{f})}{\alpha^K(\mathcal{L})} + \sum_{k\in[K]} \big(x_a^{*,k} - f_a^k\big)\big(\ell_a(f_a) + \ell'_a(f_a) f_a^k\big),$$

since the last term is nonnegative due to the variational inequality in (4.18). □

Cominetti, Correa, and Stier-Moses [23] define

$$\beta^K(\ell_a) := \sup_{\boldsymbol{f}_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \ell_a(x_a)\big)\,x_a + \ell'_a(f_a)\,\Big(\sum\limits_{k\in[K]} [f_a^k\,x_a^k - (f_a^k)^2]\Big)}{\ell_a(f_a)\,f_a}. \tag{4.21}$$

Table 4.2: Price of Anarchy for different polynomial latency functions obtained by Cominetti, Correa, and Stier-Moses [23]. All coefficients a_i are assumed to be nonnegative. The values $\alpha^\infty(\mathcal{L})$ and $\beta^\infty(\mathcal{L})$ define the values $\alpha^K(\mathcal{L})$ and $\beta^K(\mathcal{L})$ for an arbitrary number of players $K \in \mathbb{N} \cup \{\infty\}$.

Set $\mathcal{L}$ of allowable cost functions	Example	$\beta^\infty(\mathcal{L})$	Price of Anarchy $\alpha^\infty(\mathcal{L})$ arbitrary # of players
linear functions	$a_1 x + a_0$	$\frac{1}{3}$	1.5
quadratic functions	$\sum_{i=0}^{2} a_i x^i$	0.61	2.564
cubic functions	$\sum_{i=0}^{3} a_i x^i$	0.87	7.826
polynomials of degree 4	$\sum_{i=0}^{4} a_i x^i$	1.13	∞
polynomials of degree 5	$\sum_{i=0}^{5} a_i x^i$	1.38	∞
.	.		.
.	.		.
polynomials of degree d	$\sum_{i=0}^{d} a_i x^i$		∞

and $\beta^K(\mathcal{L}) := \sup_{\ell_a \in \mathcal{L}} \beta^K(\ell_a)$. This value is nonnegative, i.e., $\beta^K(\mathcal{L}) \geq 0$ and fulfills the relation $\left(1 - \beta^K(\mathcal{L})\right)^{-1} = \alpha^K(\mathcal{L})$ when $\beta^K(\mathcal{L}) < 1$. For the case $\beta^K(\mathcal{L}) \geq 1$, it is assumed that $\left(1 - \beta^K(\mathcal{L})\right)^{-1} = \infty$. This leads to the following result.

Proposition 4.18 (Roughgarden[83]). *Consider an atomic network game with K players and latency functions in $\mathcal{L}$. Let $\boldsymbol{f}$ be a Nash equilibrium and $\boldsymbol{x}^*$ be a social optimum. Then,*

$$C(\boldsymbol{f}) \leq \left(1 - \beta^K(\mathcal{L})\right)^{-1} C(\boldsymbol{x}^*).$$

Although Roughgarden [83] and Correa, Schulz, and Stier-Moses claimed indecently that the price of anarchy in the atomic case can not exceed that of the nonatomic case, it has been shown in Cominetti, Correa, and Stier-Moses [23] that this is not true. In fact they show an instance with atomic players and affine linear latency functions, where the price of anarchy is approximately 1.343. Correct upper bounds according to the results obtained by Cominetti, Correa, and Stier-Moses [23] are shown in Table 4.2.

4.4.2 Improved Bounds on the Price of Anarchy

Based on ideas of the analysis of nonatomic network games in the previous section, we introduce the parameter $\omega(\mathcal{L}, K, \lambda)$ for network games with K atomic players. The main difference between the values $\omega(\mathcal{L}, \lambda)$ and $\omega(\mathcal{L}, K, \lambda)$ is that the flow decomposition into commodities plays an important role in the latter case. The reason for this is the ability of atomic players to coordinate the flow that they control.

For every arc a, latency function ℓ_a, and nonnegative parameter λ, we define the following nonnegative value:

$$\omega(\ell_a; K, \lambda) := \sup_{f_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a + \ell'_a(f_a)\, \big(\sum_{k \in [K]} [f_a^k\, x_a^k - (f_a^k)^2]\big)}{\ell_a(f_a)\, f_a}. \tag{4.22}$$

We assume $0/0 = 1$ by convention. For a given class $\mathcal{L}$ of latency functions, we further define

$$\omega(\mathcal{L}; K, \lambda) := \sup_{\ell_a \in \mathcal{L}} \omega(\ell_a; K, \lambda).$$

Moreover, we define the following set:

Definition 4.19
The set of feasible $\lambda \geq 0$ is defined as

$$\Lambda(\mathcal{L}, K) := \{\lambda \in \mathbb{R}^+ | \big(1 - \omega(\mathcal{L}; K, \lambda)\big) > 0\}.$$

Equipped with the feasible scaling space $\Lambda(\mathcal{L}, K)$ of the parameter λ we state the main theorem.

Theorem 4.20. *For latency functions in $\mathcal{L}$, the price of anarchy for the atomic network game is at most*

$$\inf_{\lambda \in \Lambda(\mathcal{L}, K)} \left[\lambda\, (1 - \omega(\mathcal{L}; K, \lambda)^{-1})\right].$$

Proof. Let $\boldsymbol{f}$ be a flow at Nash equilibrium, and let $\boldsymbol{x}$ be any feasible flow.

$$\begin{aligned} C(\boldsymbol{f}) &\leq \sum_{a \in A} \ell_a(f_a)\, f_a + \sum_{k \in [K]} \big(\ell_a(f_a) + \ell'_a(f_a)\, f_a^k\big)(x_a^k - f_a^k) \qquad (4.23) \\ &= \sum_{a \in A} \ell_a(f_a)\, x_a + \sum_{k \in [K]} \ell'_a(f_a)\, f_a^k\, (x_a^k - f_a^k) \\ &= \sum_{a \in A} \lambda\, \ell_a(x_a)\, x_a + \big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a + \sum_{k \in [K]} \ell'_a(f_a)\, f_a^k\, (x_a^k - f_a^k) \\ &\leq \lambda\, C(\boldsymbol{x}) + \omega(\mathcal{L}; K, \lambda)\, C(\boldsymbol{f}). \qquad (4.24) \end{aligned}$$

Here, (4.23) follows from the variational inequality stated in Lemma 4.15. The last inequality (4.24) follows from the definition of $\omega(\mathcal{L}; K, \lambda)$. Taking $\boldsymbol{x}$ as the optimal offline solution and since $\lambda \in \Lambda(\mathcal{L}, K)$, the claim is proven. □

Our definition of $\omega(\mathcal{L}; K, \lambda)$ originates in a similar definition of the parameter $\beta^K(\mathcal{L})$ in Cominetti, Correa, and Stier-Moses [23] and $\alpha^K(\mathcal{L})$ in Roughgarden [83]. For a class of latency functions $\mathcal{L}$ in which $\beta^K(\mathcal{L})$ exists, we have the relation $\beta^K(\mathcal{L}) = \omega(\mathcal{L}; K, 1)$ and $\alpha^K(\mathcal{L}) = (1 - \omega(\mathcal{L}; K, 1))^{-1}$. However, neither the anarchy value $\alpha^K(\mathcal{L})$, nor the parameter $\beta^K(\mathcal{L})$ provide upper bounds for polynomial latency functions with nonnegative coefficients. Furthermore, the existing bounds derived by analyzing the value $\beta^K(\mathcal{L})$ are not known to be tight. As we show in the next section, using Theorem 4.20 it is possible to improve all previous known bounds for this class of latency functions, except for affine linear latencies.

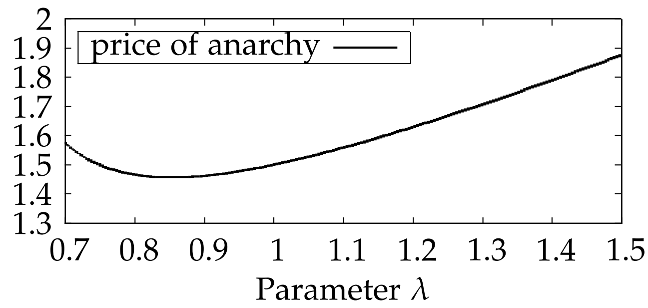

Figure 4.3: The price of anarchy for affine linear price functions as a function of the scaling parameter λ.

4.4.3 Linear and Affine Linear Latency Functions

We start with reproving a result obtained by Cominetti, Correa, and Stier-Moses [23] for the class $\mathcal{L}_1$. We present a proof for completeness showing that the best bound can be achieved by setting $\lambda = 1$. For this value of λ we have $\beta^K(\mathcal{L}_1) = \omega(\mathcal{L}_1; K, 1)$.

Theorem 4.21 (Cominetti, Correa, and Stier-Moses [23]). *Let f be a flow at Nash equilibrium of an atomic network game with latency functions in $\mathcal{L}_1$. Then,*

$$C(\boldsymbol{f}) \leq \frac{3}{2} C(\boldsymbol{x}),$$

for any feasible flow $\boldsymbol{x}$.

Proof. We assume latency functions of the form $\ell_a(z) = q_a\, z + r_a,\ q_a \geq 0,\ r_a \geq 0$. We start with the definition of $\omega(\ell_a; K, \lambda)$:

$$\omega(\ell_a; K, \lambda) = \sup_{\boldsymbol{f}_a, \boldsymbol{x}_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a + \ell'_a(f_a) \sum_{k \in [K]} (f_a^k x_a^k - f_a^k f_a^k)}{\ell_a(f_a)\, f_a}.$$

Note that we are using the notation $\boldsymbol{f}_a \geq 0$ according to (4.20). Using

$$f_a^k\, x_a^k - f_a^k\, f_a^k \leq \frac{1}{4}\, (x_a^k)^2,$$

because

$$(\frac{1}{2}\, x_a^k - f_a^k)^2 \geq 0,$$

and

$$\sum_{k \in [K]} (x_a^k)^2 \leq (x_a)^2,$$

we get the following bound:

$$\omega(\ell_a; K, \lambda) \leq \sup_{f_a, x_a \geq 0} \frac{\big(q_a\, f_a - \lambda\, q_a\, x_a\big)\, x_a + \big(r_a - \lambda\, r_a\big)\, x_a + q_a\, \frac{(x_a)^2}{4}}{q_a\, (f_a)^2 + r_a\, f_a}.$$

Here, we use $f_a \geq 0$ indicating that the flow decomposition into commodities becomes irrelevant. The term $\big(r_a - \lambda\, r_a\big)\, x_a$ inside the supremum leads to the condition $\lambda \geq 1$, since otherwise we can set $q_a = 0$ and let x_a tend to infinity to make the supremum unbounded. For $\lambda \geq 1$ we can simplify the supremum.

$$\omega(\ell_a; K, \lambda) \leq \sup_{f_a, x_a \geq 0} \frac{\big(f_a - \lambda\, x_a\big)\, x_a + \frac{(x_a)^2}{4}}{(f_a)^2} = \max_{\mu \geq 0} \big(\mu - \lambda\, \mu^2 + \frac{\mu^2}{4}\big),$$

where $\mu := \frac{x_a}{f_a}$ if $f_a > 0$ and $\mu = 0$ otherwise. The unique optimal solution is given by $\mu^* = \frac{2}{4\lambda - 1}$. Inserting this value into the objective leads to

$$\omega(\ell_a; K, \lambda) \leq \frac{1}{4\,\lambda - 1}. \tag{4.25}$$

Applying Theorem 4.20 yields:

$$C(\boldsymbol{f}) \leq \min_{\lambda \geq 1} \frac{\lambda}{1 - \frac{1}{4\lambda - 1}}\, C(\boldsymbol{x}) = \min_{\lambda \geq 1} \lambda\, \frac{4\,\lambda - 1}{4\,\lambda - 2}\, C(\boldsymbol{x}) = \frac{3}{2}\, C(\boldsymbol{x}),$$

where the optimal value is $\lambda^* = 1$. □

For purely linear latency functions, i.e. latencies in $\mathcal{M}_1$, we can further improve the best known bound of $\frac{3}{2}$ by varying λ below 1. The function $\frac{4\lambda - 1}{4\lambda - 2}$ is plotted in Figure 4.3.

Corollary 4.22. *Consider latency functions in $\mathcal{M}_1$. Then, the price of anarchy is at most* $\frac{1}{8}\,(2 + \sqrt{2})\,(1 + \sqrt{2})\,\sqrt{2} \approx 1.46.$

Proof. We can start with inequality (4.25). Analyzing the feasible set $\Lambda(\mathcal{M}_1, K)$ we get $\frac{1}{4\lambda - 1} < 1$, which is equivalent to $\lambda > \frac{1}{2}$. Applying Theorem 4.20 yields:

$$C(\boldsymbol{f}) \leq \min_{\lambda > \frac{1}{2}} \lambda\, \frac{4\,\lambda - 1}{4\,\lambda - 2}\, C(\boldsymbol{x}) = \frac{1}{8}\,(2 + \sqrt{2})\,(1 + \sqrt{2})\,\sqrt{2}\, C(\boldsymbol{x}),$$

where we set $\lambda = \frac{1}{2} + \frac{1}{4}\,\sqrt{2}$.

□

4.4.4 Lower Bounds

In the following, we present a lower bound on the price of anarchy for purely linear latency functions. These bounds demonstrate that in contrast to the nonatomic counterpart the price of anarchy may be larger than 1 for linear latency functions. The following instance is taken from Cominetti, Correa, and Stier-Moses [23].

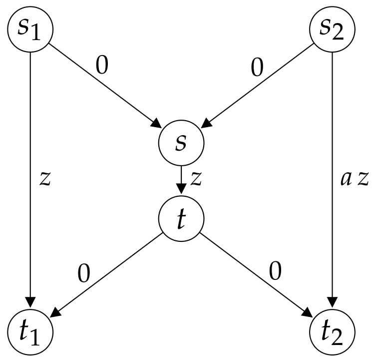

Figure 4.4: Graph construction for the proof of Proposition 4.23.

Proposition 4.23. *In case of linear latency functions, the price of anarchy for the atomic network routing game is bounded from below by* $1\,\frac{1}{25}$.

Proof. Consider the network given in Figure 4.4. Note that all latency functions have $\ell_a(0) = 0$. We assume that a nonatomic player (N) wants to route one unit from node s_1 to node t_1. On the other hand, one atomic player (A) wants to route one unit from s_2 to node t_2. For both players, there exist possible paths: the direct path (s_1, t_1) and (s_2, t_2) or the path along the shared arc (s, t). If x and y denote the amount of flow for player N, and player A, that is routed along the direct arc (s_1, t_1), and (s_2, t_2), respectively. The response strategies are given by the following two optimization problems. For player N we have:

$$\min_{0 \le x \le 1} \frac{1}{2}\,x^2 + \frac{1}{2}\,(1-x)^2 + (1-x)\,(1-y). \tag{4.26}$$

Note that is assumed that player A sends $1-y$ units flow along the middle arc. Hence, $\ell_{(s,t)}(z + (1-y)) = z + (1-y)$. The optimal solution to problem (4.26) is given by

$$x^* = \min\left\{\max\left\{\frac{2-y}{2}, 0\right\}, 1\right\}.$$

For player A we have:

$$\min_{0 \le y \le 1} a\,y^2 + ((1-x) + (1-y))(1-y). \tag{4.27}$$

The solution is given by

$$y^* = \min\left\{\max\left\{\frac{3-x}{2\,a+2}, 0\right\}, 1\right\}.$$

Plugging both solutions together and assuming $\frac{1}{2} \le a$ yields:

$$y^* = \frac{4}{4\,a+3}, \text{ and } x^* = \frac{4\,a+1}{4\,a+3}.$$

If we denote the entire flow by f, then the cost in equilibrium is given by

$$C(\boldsymbol{f}) = \frac{32\,a^2 + 32\,a + 2}{(4\,a+3)^2}.$$

Now the optimal flow $\boldsymbol{x}^*$ solves:

$$\min_{\substack{0 \le x \le 1 \\ 0 \le y \le 1}} x^2 + \big((1-x) + (1-y)\big)^2 + a\,y^2. \tag{4.28}$$

Here, the optimal solutions are given by

$$y^* = \frac{2}{2\,a+1}, \text{ and } x^* = \frac{2\,a}{2\,a+1}.$$

$$C(\boldsymbol{x}^*) = \frac{4\,a}{2\,a+1}.$$

Setting $a := \frac{1}{2}$ yields

$$C(\boldsymbol{f}) = \frac{26}{25}, \text{ and } C(\boldsymbol{x}^*) = 1,$$

proving the claim. □

Optimizing over the paramers used in the above example, we are able to raise the bound to 1.17. This bound has already been established by Cominetti, Correa and Stier-Moses [23].

4.4.5 General Latency Functions

We start this section with bounding the value $\omega(\ell_a; K, \lambda)$ for s-convex latency functions. Some of the following results (Lemma 4.24, Lemma 4.25, and Proposition 4.26) are based on results obtained by Cominetti, Correa and Stier-Moses [23]. Even though the proofs are almost identical, we need to keep track of restrictions on the parameter λ. For this reason, we also present complete proofs.

Lemma 4.24. *Assume that $\lambda \ge 0$ and $\ell_a(\cdot)$ is a continuous nondecreasing latency function. Then, the following inequality is valid:*

$$\omega(\ell_a; K, \lambda) \le \sup_{f_a, x_a \ge 0} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)\,x_a + \ell_a'(f_a)\,\frac{x_a^2}{4}}{\ell_a(f_a)\,f_a}. \tag{4.29}$$

Proof. We start with the definition of $\omega(\ell_a; K, \lambda)$:

$$\omega(\ell_a; K, \lambda) = \sup_{x_a, f_a \ge 0} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)x_a + \ell_a'(f_a)\Big(\sum_{k \in [K]} (f_a^k\,x_a^k - f_a^k\,f_a^k)\Big)}{\ell_a(f_a) f_a}.$$

First, we bound the last difference in the numerator:

$$f_a^k\, x_a^k - f_a^k\, f_a^k \leq \frac{1}{4}\,(x_a^k)^2,$$

since

$$(\frac{1}{2}\,x_a^k - f_a^k)^2 \geq 0.$$

This yields:

$$\omega(\ell_a; K, \lambda) \leq \sup_{x_a, f_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)x_a + \ell_a'(f_a)\Big(\sum_{k\in[K]} \frac{(x_a^k)^2}{4}\Big)}{\ell_a(f_a) f_a}.$$

Finally, using $\sum_{k\in[K]} (x_a^k)^2 \leq (x_a)^2$ and $\ell_a'(f_a) \geq 0$ proves the lemma. □

We define $\omega(\ell_a; \infty, \lambda)$ to be the limit of $\omega(\ell_a; K, \lambda)$ for K tending to infinity under the condition that x_a and f_a are kept constant (and hence $\omega(\ell_a; K, \lambda)$ stays finite). Then it follows that $\omega(\ell_a; K, \lambda) \leq \omega(\ell_a; \infty, \lambda)$. We focus in the following on the general case $K \in \mathbb{N} \cup \{\infty\}$.

Lemma 4.25. *If $\lambda \geq 1$ and $\ell_a(f_a)\, f_a$ is a convex function, then the value $\omega(\ell_a; \infty, \lambda)$ is at most:*

$$\omega(\ell_a; \infty, \lambda) \leq \sup_{0 \leq x_a \leq f_a} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)\,x_a + \ell_a'(f_a)\,\frac{(x_a)^2}{4}}{\ell_a(f_a)\,f_a}. \tag{4.30}$$

Proof. Consider the function $h(x_a)$ defined as the numerator of the supremum in (4.29). To prove that the solution satisfies $x_a \leq f_a$, we show that $h'(x_a) \leq 0$ if $x_a \geq f_a$. Using that $h'(x_a) = \ell_a(f_a) - \lambda\,\ell_a(x_a) - \lambda\,x_a\,\ell_a'(x_a) + \frac{x_a}{2}\,\ell_a'(f_a)$, the derivative is negative if and only if

$$\ell_a(f_a) + \frac{x_a}{2}\,\ell_a'(f_a) \leq \lambda\,\big(\ell_a(x_a) + x_a\,\ell_a'(x_a)\big).$$

By assumption $\ell_a(f_a)\,f_a$ is convex, hence,

$$\ell_a(f_a) + \ell_a'(f_a)\,f_a \leq \ell_a(x_a) + \ell_a'(x_a)\,x_a$$

for $x_a \geq f_a$. Since furthermore $\lambda \geq 1$, the proof is complete. □

The following characterization of $\omega(\ell_a; K, \lambda)$ via a continuous and differentiable function $s : [0,1] \to [0,1]$ is based on ideas of Cominetti, Correa, and Stier-Moses [23].

Proposition 4.26. *Let $\mathcal{L}$ be a class of continuous, nondecreasing, and convex latency functions $\ell_a(\cdot)$. Furthermore, assume that $\lambda \geq 1$ and $\ell_a(\kappa\,f_a) \geq s(\kappa)\,\ell_a(f_a)$ for all $\kappa \in [0,1]$, where $s : [0,1] \to [0,1]$ is a differentiable function with $s(1) = 1$. Then,*

$$\omega(\ell_a; \infty, \lambda) \leq \max_{0 \leq u \leq 1} u\,\big(1 - \lambda\,s(u) + s'(1)\,\frac{u}{4}\big). \tag{4.31}$$

Proof. We start with the characterization of $\omega(\ell_a;\infty,\lambda)$ given in Lemma 4.24:

$$\omega(\ell_a;\infty,\lambda) \le \sup_{f_a,x_a\ge 0} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)\,x_a + \ell_a'(f_a)\Big(\frac{x_a^2}{4}\Big)}{\ell_a(f_a)\,f_a}.$$

For $z \ge z'$, we can bound $\ell_a'(z)$:

$$\ell_a(z') = \ell_a(\tfrac{z'}{z}\,z) \ge s(\tfrac{z'}{z})\,\ell_a(z)\,z. \tag{4.32}$$

Furthermore,

$$\ell_a'(f) = \lim_{\epsilon\to 0}\frac{\ell_a(f_a+\epsilon)-\ell_a(f_a)}{\epsilon} \le \ell_a(f_a)\lim_{\epsilon\to 0}\frac{1-s\big(\frac{f_a}{f_a+\epsilon}\big)}{\epsilon} = \ell_a(f_a)\,\frac{s'(f_a)}{f_a}.$$

Thus, we conclude

$$\begin{aligned}\omega(\ell_a;\infty,\lambda) &\le \sup_{0\le x_a\le f_a} \frac{x_a\,\ell_a(f_a)\big(1-\frac{\lambda\,\ell_a(x_a)}{\ell_a(f_a)}+\frac{s'(1)\,x_a}{4\,f_a}\big)}{\ell_a(f_a)\,f_a}\\ &\le \sup_{0\le x_a\le f_a}\frac{x_a\,\big(1-\lambda\,s(\frac{x_a}{f_a})+\frac{s'(1)\,x_a}{4\,f_a}\big)}{f_a},\end{aligned}$$

where we used (4.32) for the second inequality. Defining $0 \le u := \frac{x_a}{f_a} \le 1$ yields

$$\omega(\ell_a;\infty,\lambda) \le \max_{0\le u\le 1} u\,\big(1-\lambda\,s(u)+s'(1)\,\frac{u}{4}\big).$$

□

Corollary 4.27. *If latency functions are in $\mathcal{L}_d$, $d \ge 1$, the price of anarchy is at most*

$$\inf_{\lambda\in\Lambda(\mathcal{L}_d,K)\cap\mathbb{R}^{\ge 1}}\left[\lambda\left(1-\max_{0\le u\le 1} u\,\big(1-\lambda\,u^d+d\,\frac{u}{4}\big)\right)^{-1}\right]. \tag{4.33}$$

Proof. All assumptions of Proposition 4.26 are satisfied with $s(f) = f^d$. Therefore, $s'(1) = d$ and

$$\omega(\ell_a;\infty,\lambda) \le \max_{0\le u\le 1} u\,\big(1-\lambda\,u^d+d\,\frac{u}{4}\big). \tag{4.34}$$

Applying Theorem 5.34 yields the claim. □

In Table 4.3, we present results for squared, cubic, and degree four, and five polynomials. Note that all results improve previously known bounds, except for the affine linear case. The results itself have been obtained by optimizing the expression in (4.33) over the parameter $\lambda \in \Lambda(\mathcal{L}_d,K)\cap\mathbb{R}^{\ge 1}$.

Theorem 4.28. *If latency functions are in $\mathcal{L}_d$, $d \ge 1$, then, the price of anarchy is at most $\big(1+\frac{d}{4}\big)^{d+1}$.*

Table 4.3: Upper and lower bounds on the price of anarchy for network games with atomic players. Considered are polynomial latency functions. Coefficients a_i are assumed to be nonnegative. The bound 1.5 for affine latency functions and the lower bounds for affine linear and linear latencies are due to Cominetti, Correa and Stier-Moses [23]. The lower bounds for degree larger than 1 are the matching lower bounds for nonatomic network games. The value $\omega(\mathcal{L};\infty,\lambda)$ is defined in (4.22).

Set $\mathcal{L}$ of allowable latency functions	Example	$\omega(\mathcal{L};\infty,\lambda)$	λ	UB arbitrary	LB # of players
linear functions	$a_1 x$	0.41	0.85	1.46	1.17 [23]
affine functions	$a_1 x + a_0$	$\frac{1}{3}$	1	1.5 [23]	1.34 [23]
quadratic functions	$\sum_{i=0}^{2} a_i x^i$	0.58	1.08	2.55	1.63 [81]
cubic functions	$\sum_{i=0}^{3} a_i x^i$	$\frac{2}{3}$	1.69	5.06	1.90 [81]
polynomials $d \le 4$	$\sum_{i=0}^{4} a_i x^i$	$\frac{2}{3}$	3.8	11.3	2.15 [81]
polynomials $d \le 5$	$\sum_{i=0}^{5} a_i x^i$	$\frac{2}{3}$	9.69	29.07	2.39 [81]

Proof. We start the proof by bounding the value $\omega(\ell_a;\infty,\lambda)$ from above. Recall from Equation (4.34) that

$$\omega(\ell_a;\infty,\lambda) \le \max_{0\le u\le 1} u\left(1-\lambda\, u^d + d\,\frac{u}{4}\right).$$

Setting $u = 1$ in the last term yields

$$\omega(\ell_a;\infty,\lambda) \le \max_{0\le u\le 1} u\left(1-\lambda\, u^d + \frac{d}{4}\right).$$

This problem is a standard concave program on a compact interval. Hence, it admits a solution. For $d \ge 1$ the objective is strictly concave implying that the solution is unique. The necessary and sufficient optimality condition for a global optimum that satisfies $u \in (0,1)$ is given by

$$1 + \frac{d}{4} - (d+1)\,\lambda\, u^d = 0.$$

Hence, the optimal solution is given by

$$u^* = \min\left\{\max\left\{\left(\frac{4+d}{4\,\lambda\,(d+1)}\right)^{\frac{1}{d}},0\right\},1\right\}.$$

We assume $1 \le \lambda < \infty$ which implies $0 < u^* = \frac{4+d}{4\,\lambda\,(d+1)}^{\frac{1}{d}} < 1$. Inserting this solution into the objective leads to

$$\omega(\ell_a;\infty,\lambda) \le \left(\frac{4+d}{4\,\lambda\,(d+1)}\right)^{\frac{1}{d}}\left(\frac{4\,d+d^2}{4\,(d+1)}\right).$$

We construct a function $1 \leq \lambda(d) < \infty$ such that for all $d \geq 1$ the following equation holds

$$\left(\frac{4+d}{4\,\lambda(d)\,(d+1)}\right)^{\frac{1}{d}} \left(\frac{4\,d+d^2}{4\,(d+1)}\right) = \frac{d}{d+1}.$$

Solving the above equation with respect to $\lambda(d)$ yields

$$\lambda^*(d) = \frac{(4+d)^{d+1}}{(d+1)\,4^{d+1}}.$$

Thus, by construction we have

$$\omega(\ell_a;\infty,\lambda^*(d)) \leq \frac{d}{d+1}.$$

Applying Theorem 5.34 with $\lambda := \lambda^*(d)$ and $\omega(\ell_a;\infty,\lambda^*(d)) \leq \frac{d}{d+1}$ leads to

$$C(\boldsymbol{f}) \leq \frac{\lambda^*(d)}{1-\frac{d}{d+1}}\,C(\boldsymbol{x}) = (d+1)\,\lambda^*(d)\,C(\boldsymbol{x}) = \big(1+\frac{d}{4}\big)^{d+1}\,C(\boldsymbol{x}).$$

□

Note that a similar technique can be applied to strengthen the bounds on the price of anarchy. The idea is to construct a function $\lambda(d)$ such that

$$\max_{0\leq u\leq 1} u\,\big(1-\lambda(d)\,u^d + d\,\frac{u}{4}\big) = \frac{2}{3}$$

holds for all $d \geq 1$. Then, the price of anarchy can be bounded by $3\,\lambda(d)$. The function $\lambda(d)$ behaves asymptotically like $O\big(\exp(\frac{2}{5}\,\log(d))\big)$.

Chapter 5

Online Network Games

The network games presented in the previous chapter are special cases of congestion games introduced by Rosenthal [78]. In a congestion game, players select strategies that are subsets of resources, and the utility of a player only depends on the number of players choosing the same or some overlapping strategy.

A direct application of network games is the source routing concept in telecommunication networks, see Qiu, Yang, Zhang, and Shenker [76] and Friedman [42] for an engineering perspective and Roughgarden [80] and Altman, Basar, Jimenez, and Shimkin [5] for a theoretical perspective on this topic. In the source routing model, sources are responsible for selecting paths to route data to the corresponding sink. The links in the network advertise their current status (price) that is based on the current congestion situation. If the link prices correspond to the expected delay on that link, minimum cost routing is a natural goal for time critical real-time applications. If sources select routes based on such selfish interests, the flows converge to a Nash equilibrium, as observed by Qiu, Yang, Zhang, and Shenker [76]. The main focus of the research done so far regarding the source routing concept is to quantify the efficiency loss of a Nash equilibrium compared to the system optimum. Here, one assumption is crucial: if the traffic matrix changes, all sources may possibly change their routes and converge to a new equilibrium, see Even-Dar and Mansour [31] for a further discussion about the convergence behavior. This assumption, however, has some important implications: Each source would have to *continuously* maintain the current state of all available routes, which in turn introduces additional traffic overhead by signaling these needed informations. Furthermore, frequent rerouting attempts during data transmission may not only produce transient load oscillations but may also interfere with the widely used congestion control protocol TCP that determines the data rate, as reported by La, Walrand, and Anantharam in [63]. For these reasons, frequent rerouting attempts in reaction to traffic changes in the network are not necessarily beneficial and efficient. Time critical applications, such as Internet Telephony or video streaming may suffer severe performance degradation.

To overcome some of the above stated problems, we investigate in this chapter a new model in which sources starting at the same time select their routes *only* during a predefined time frame. In this model, we assumes that sources starting within the same time frame converge to an equilibrium before new sources appear. Then, we investigate the extreme case in which flows once they are at equilibrium fix their routing decisions. Thus, continuously gathering information about the current network state is not necessary after this initial routing game. We can interpret this model as follows. We introduce a cost for each player quantifying the cost of rerouting after the time frame. If this rerouting cost is sufficiently large for each player, then, fixing the initial equilibrium routing is the best response strategy.

To analyze this model we introduce the concept of *online network games*. In this concept, we assume a sequence of network games $\sigma = 1, \dots, n$ that are released consecutively in time in an online fashion. By the time of releasing game i, future games $i+1, \dots, n$ are not known. We assume that once commodities of a game are routed, they remain fixed. We analyze two online algorithms, called NSeqNash and ASeqNash in this setting. These algorithms produce a flow consisting of a sequence of Nash equilibria for the corresponding games with nonatomic and atomic players, respectively. As usual, we analyze the efficiency of an online algorithm in terms of competitive analysis. The optimal offline solution in our model is derived by minimizing the total routing cost for all games. The total routing cost is defined as in Chapter 4 by summing over all arc costs. The cost of an arc is defined as the product of latency and flow on that arc. Note that for deriving the optimal offline solution, the sequence σ is known a priori. It turns out that a combination of the online optimization field with algorithmic game theory provides a fruitful way to analyze the efficiency of NSeqNash and ASeqNash in this framework. The main result in this chapter states that the inefficiency of the sequence of Nash equilibria can be bounded by a constant factor for polynomial latency functions with nonnegative coefficients regardless of the player types. Although the constants in general are large, these results indicate that the above routing model does not lead to situations that are arbitrary far from the best possible situation. We are aware that some of our assumptions are quite restrictive. Nevertheless, we believe that our model approximates the dynamics of a real system such as the Internet. In this regard, we interpret our results as a first step towards understanding the dynamic behavior of network flows beyond the single static equilibrium concept.

The online model in this chapter is closely related to the model in Chapter 3, see also Harks, Heinz, and Pfetsch [52], where online multicommodity routing problems are considered. There, however, we studied a greedy online algorithm for a different convex cost function. Recall that in the OnlineMCRP the cost for a commodity is independent of the routing of later commodities even if later commodities use the same arcs than the former commodity. In online network games, this is not the case. Routing decisions of commodities in later games may affect the cost of commodities of previous games if the chosen routes have overlapping arcs. Furthermore, for a given sequence of games, the online algorithm that produces a flow at Nash equilibrium for every game is not of greedy type. Consider for instance a sequence σ with a single game, or equivalently, a single commodity in the OnlineMCRP setting. For the OnlineMCRP it is easy to show that Seq is optimal. A flow at Nash equilibrium, however, is known to be inefficient for most instances. Very recent, Engelberg and Naor [30] drew connections between online optimization and algorithmic game theory. In their framework, they present different examples in which a player has to choose an online algorithm in order to minimize its individual competitive ratio. Work on the convergence behavior of flows for the parallel link setting can be found in Even-Dar and Mansour [31] and Fischer and Vöcking [33] and Fischer, Räcke, and Vöcking [34]. Further work on convergence to a Nash equilibrium for a setting in which flows sequentially join the network can be found in Blum, Even-Dar, and Ligett [14]. None of these works, however, analyze the efficiency of flows arriving sequentially without adapting to the common static Nash equilibrium.

5.1 Contributions and Chapter Outline

We introduce the framework *Online Network Games* (OnlineNG) to analyze online routing problems. For the online algorithm NSeqNash that is characterized by selfish routing of *nonatomic* players for a sequence of network games, we obtain the following results. The online algorithm NSeqNash that produces a flow that is at Nash equilibrium for every game is $\frac{4n}{2+n}$ -competitive for affine linear latency functions, where n is the number of games within a given sequence. This result contains the bound on the price of anarchy of $\frac{4}{3}$ for affine linear latency functions of Roughgarden and Tardos [84] as a special case of our model, where $n = 1$. We prove a lower bound of $\frac{3n-2}{n}$ of nonatomic NSeqNash showing that for $n = 2$, the upper bound is tight. For linear latency functions, we further improve this bound to $\frac{4n^2}{(1+n)^2}$. For polynomial latency functions with nonnegative coefficients, we prove lower and upper bounds on the competitive ratio of NSeqNash that grow both exponentially in the degree of the considered polynomials. We further show that for parallel arcs, the competitive ratio is significantly lower. In particular, we show that in this case the competitive ratio of the NSeqNash does not exceed the price of anarchy of a complementary nonatomic network game in which all games of a given sequence are considered at the same time.

Furthermore, we consider the online algorithm ASeqNash that models the selfish behavior of atomic players. Our main results for this variant are summarized in the following. The online algorithm ASeqNash that produces a flow that is at Nash equilibrium for every game within a given sequence of games is $\min\{\frac{2(3\mathcal{K}+1)n}{n\mathcal{K}+3n+3\mathcal{K}+1}, \frac{5\mathcal{K}+1}{\mathcal{K}+5}, 4.92\}$-competitive for affine linear latency functions, where $\mathcal{K}$ denotes the total number of players and n is the number of games within a given sequence. For general polynomial latency functions, we prove lower and upper bounds on the competitive ratio of ASeqNash that grow both exponentially in the degree of the considered polynomials.

The chapter is organized as follows. In Section 5.2, we introduce the basic model of online network games. Then, in Section 5.3, we study the efficiency of NSeqNash for a sequence of network games with nonatomic players. We extend the analysis in Section 5.4 to online network games with atomic players.

The results of this chapter are joined work with L. A. Végh and appear in [51].

5.2 Online Network Games

An instance of the *Online Network Game* (OnlineNG) consists of a directed network $D = (V, A)$ together with nondecreasing continuous latency functions $\ell_a : \mathbb{R}_+ \to \mathbb{R}_+$ for each arc $a \in A$. Furthermore, a sequence $\sigma = 1, \dots, n$ of network games are given. A network game i is characterized by a set of commodities $[K_i] := \{i1, \dots, i\,n_i\}$. For each commodity $ij \in [K_i]$, a flow of rate $d_{ij} > 0$ must be routed from the origin s_{ij} to the destination t_{ij}. The routing decision for game i is *online*, that is, it only depends on the routings of previous games $1, \dots, i-1$. Once the commodities of a game have been routed they remain unchanged. Let $[\mathcal{K}] = \bigcup_{i=1}^{n} [K_i]$ denote the union of the sets $[K_1], \dots, [K_n]$. The total number of commodities is given by $\mathcal{K} = \sum_{i=1}^{n} n_i$.

A routing assignment, or *flow*, for commodity $ij \in [K_i]$ is a nonnegative vector $f^{ij} \in \mathbb{R}_+^A$. This flow is *feasible*, if for all $v \in V$

$$\sum_{a \in \delta^+(v)} f_a^{ij} - \sum_{a \in \delta^-(v)} f_a^{ij} = \gamma_{ij}(v), \tag{5.1}$$

where $\delta^+(v)$ and $\delta^-(v)$ are the arcs leaving and entering v, respectively; furthermore,

$$\gamma_{ij}(v) = \begin{cases} d_{ij}, \text{ if } v = s_{ij}, \\ -d_{ij}, \text{ if } v = t_{ij}, \\ 0, \text{ otherwise.} \end{cases} \tag{5.2}$$

Alternatively, one can consider a *path flow* for a commodity $ij \in [K_i]$. Let $\mathcal{P}_{ij}$ be the set of all paths from s_{ij} to t_{ij} in D. A path flow is a nonnegative vector $(f_P^{ij})_{P \in \mathcal{P}_{ij}}$. The corresponding flow on link $a \in A$ for commodity $ij \in [K_i]$ is then

$$f_a^{ij} := \sum_{P \ni a} f_P^{ij}.$$

We denote by

$$f_a^i = \sum_{ij \in [K_i]} f_a^{ij}$$

the aggregated flow of game i on link a. The total aggregate flow on link a is given by

$$f_a = \sum_{i=1}^{n} f_a^i.$$

We define $\mathcal{F}_i$ with $i \in [n]$ to be the set of vectors $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^i)$ such that $\boldsymbol{f}^j$ is a feasible flow for games $j = 1, \ldots, i$. If $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^i) \in \mathcal{F}_i$, we say that it is *feasible* for the sequence of network games $1, \ldots, i$. The entire flow for the sequence $1, \ldots, n$ of games is denoted by $\boldsymbol{f} = (\boldsymbol{f}^1, \ldots, \boldsymbol{f}^n)$.

The *current* cost of a feasible flow for game i on link $a \in A$ is defined by

$$C_a^i(\boldsymbol{f}_a^i; \boldsymbol{f}_a^1, \ldots, \boldsymbol{f}_a^{i-1}) := \ell_a\Big(\sum_{j=1}^{i} f_a^j\Big) f_a^i.$$

This expression can be obtained as the routing cost on arc a for a feasible flow for game i, *given* the flows $(\boldsymbol{f}^1, \ldots, \boldsymbol{f}^{i-1})$ of previous games $1, \ldots, i-1$ and *without* knowing about future games $j = i+1, \ldots, n$. The individual current cost for commodity $ij \in [K_i]$ on arc a is given by

$$C_a^{ij}(\boldsymbol{f}_a^i; \boldsymbol{f}_a^1, \ldots, \boldsymbol{f}_a^{i-1}) = \ell_a\Big(\sum_{j=1}^{i} f_a^j\Big) f_a^{ij}.$$

Note that this individual current cost on arc a may increase if later commodities are routed on a. The current cost for game i is given by the sum of arc costs

$$C^i(\boldsymbol{f}^i; \boldsymbol{f}^1, \ldots, \boldsymbol{f}^{i-1}) = \sum_{a \in A} C_a^i(\boldsymbol{f}_a^i; \boldsymbol{f}_a^1, \ldots, \boldsymbol{f}_a^{i-1}).$$

The *total cost* on arc a is defined as

$$C_a(f_a) = \ell_a(f_a)\, f_a.$$

The *total cost* of all sequentially played games is given by:

$$C(\boldsymbol{f}) = \sum_{a \in A} C_a(f_a) = \sum_{a \in A} \ell_a(f_a)\, f_a = \sum_{a \in A} \ell_a\Big(\sum_{i=1}^{n} f_a^i\Big)\Big(\sum_{i=1}^{n} f_a^i\Big). \tag{5.3}$$

This cost function reflects the routing cost provided all commodities of the entire sequence of games have been routed. Thus, the cost of routing commodities of a sequence of games is not separable with respect to the games. That is, if an online algorithm routes flow for the games $i+1, \ldots, n$ along arcs that are used by commodities of games $1, \ldots, i$, the latter commodities may face higher individual cost on these arcs compared to their current routing costs.

5.2.1 Player Types

Motivated by the source routing model in communication networks, we focus in this chapter on selfish behavior of *players* routing the demands d_{ij}, $ij \in [\mathcal{K}]$. In the following, we use the word commodity ij interchangeably with *player* ij to indicate that this player decides on the routing assignment $\boldsymbol{f}^{ij}$ for the demand d_{ij}. In the nonatomic routing variant, we assume infinitely many agents

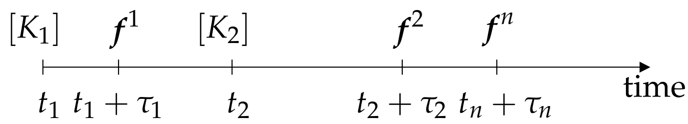

Figure 5.1: Illustration of the applicability of the considered online model to the source routing concept in the Internet. The times t_1 and t_2 are the release times of the sets of commodities $[K_1]$ and $[K_2]$ and the values τ_1, τ_2 are the times to converge to the corresponding Nash equilibrium.

carrying the flow of a player, where each agent controls only an infinitesimal fraction of the flow. This is in contrast to the atomic routing variant, where it is assumed that each player ij controls and coordinates the entire flow for his demand d_{ij}. For a sequence of games, we investigate in this chapter the online algorithm NSeqNash and ASeqNash (a formal definition follows), which produce a sequence of feasible flows $f^1, \ldots, f^n \in \mathcal{F}_n$, where each f^i is at Nash equilibrium for the corresponding network game i. We focus on the *efficiency* of NSeqNash and ASeqNash compared to the offline optimum Opt using competitive analysis coming from the online optimization field. Throughout the chapter we assume that splitting of flow is allowed for every commodity. Figure 5.1 describes the needed assumptions for modeling the outcome of distributed selfishly behaving users by such online algorithms. We assume that users arrive in groups $[K_i]$ and converge to a Nash equilibrium within time τ_i before new groups arrive. If no player is willing to reroute its flow even if the traffic on used arcs changes, this yields an online algorithm that we call NSeqNash and ASeqNash for nonatomic and atomic players, respectively.

5.2.2 Nash Equilibria for Nonatomic Players

A flow for game i is at Nash equilibrium, if no player has an incentive to unilaterally change his strategy. We assume that players of game i decide on their strategies without taking future games $j = i+1, \ldots, n$ into account. It is straight-forward to check that a Nash flow f^i for nonatomic players is the optimum of the following convex optimization problem, see for example Roughgarden and Tardos [84].

$$
\begin{aligned}
\min \quad & \sum_{a \in A} \int_0^{f_a^i} \ell_a\Big(\sum_{k=1}^{i-1} f_a^k + z\Big)\, dz && && (5.4)\\
\text{s.t.} \quad & \sum_{a \in \delta^+(v)} f_a^{ij} - \sum_{a \in \delta^-(v)} f_a^{ij} = \gamma_{ij}(v) && \forall\, v \in V,\ ij \in [K_i] \\
& f_a^{ij} \geq 0 && \forall\, a \in A,\ ij \in [K_i],
\end{aligned}
$$

where $\gamma_{ij}(v)$ is defined as in (5.1). The following conditions are necessary and sufficient to characterize a Nash equilibrium for game i.

Lemma 5.1. *A feasible flow $\boldsymbol{f}^i$ for the nonatomic game i is at Nash equilibrium if and only if it satisfies:*

$$\sum_{a \in A} \ell_a\Big(\sum_{k=1}^{i} f_a^k\Big)\,(f_a^i - x_a^i) \leq 0 \text{ for all feasible flows } \boldsymbol{x}^i \text{ for game } i. \tag{5.5}$$

The proof is based on the first order optimality conditions and the convexity of the objective in (5.4), see Dafermos and Sparrow [27].

Definition 5.2 (NSeqNash for the OnlineNG)
Consider an instance of the OnlineNG with a given sequence σ of n network games. The deterministic online algorithm NSeqNash produces a feasible flow denoted by $\boldsymbol{f} = (\boldsymbol{f}^1, \ldots, \boldsymbol{f}^n) \in \mathcal{F}_n$, such that each flow $\boldsymbol{f}^k$ solves problem (5.4), that is, each $\boldsymbol{f}^k$ is at Nash equilibrium for the corresponding games $k \in [n]$.

Note that problem (5.4) is well defined and admits an optimal solution with a unique objective value. Hence, NSeqNash is also well defined by this property. Since, problem (5.4) may have several different solutions (with the same objective value), the flow that NSeqNash produces is not necessarily unique. As this might contradict the notion of a *deterministic* online algorithm, we can advise a selection rule to make the flow unique. We omit this issue in the following, since our results hold for *every* sequence of Nash flows for the games $1, \ldots, n$.

5.2.3 Nash Equilibria for Atomic Players

In network games with atomic players, some players may control a significant part of the entire demand. In the following, we characterize the strategy of an atomic player. A flow for game i is at Nash equilibrium when no player ij has an incentive to unilaterally change his strategy. It is straightforward to see that a best reply strategy for player ij of game i is to solve the following convex optimization problem.

$$\begin{aligned} \min \quad & \sum_{a \in A} \ell_a\Big(\sum_{j=1}^{i} f_a^j\Big) f_a^{ij} \\ \text{s.t.} \quad & \sum_{a \in \delta^+(v)} f_a^{ij} - \sum_{a \in \delta^-(v)} f_a^{ij} = \gamma_{ij} \quad \forall\, v \in V,\ ij \in [K_i] \\ & f_a^{ij} \geq 0 \qquad\qquad\qquad\qquad\qquad \forall\, a \in A,\ ij \in [K_i], \end{aligned} \tag{5.6}$$

where γ_{ij} is defined as in (5.2). The following conditions are necessary and sufficient to characterize a Nash equilibrium for game i.

Lemma 5.3. *A feasible flow $\boldsymbol{f}^i$ for the game i is at Nash equilibrium if and only if for every player $ij \in [K_i]$ the following inequality is satisfied:*

$$\sum_{a\in A}\Big(\ell_a\big(\sum_{k=1}^{i} f_a^k\big) + \ell_a'\big(\sum_{k=1}^{i} f_a^k\big) f_a^{ij}\Big)\,(f_a^{ij} - x_a^{ij}) \le 0, \tag{5.7}$$

for all feasible flows $\boldsymbol{x}^{ij}$ for game i.

The proof relies on the convexity of $\ell_a(z)\,z$. See also the proof of Lemma 3.4 in Chapter 3.

Definition 5.4 (ASeqNash for the OnlineNG)
Consider an instance of the OnlineNG with a given sequence σ of n network games. The deterministic online algorithm ASeqNash produces a feasible flow denoted by $\boldsymbol{f} = (\boldsymbol{f}^1, \ldots, \boldsymbol{f}^n) \in \mathcal{F}_n$, such that each flow $\boldsymbol{f}^{ij}$, $ij \in [K_i]$, $i \in [n]$ solves problem (5.6), that is, each $\boldsymbol{f}^i$ is at Nash equilibrium for the corresponding games $i \in [n]$.

Since we assume s-convex latency functions, problem (5.6) is well defined and admits an optimal solution with a unique objective value. Then, the existence of a flow at Nash equilibrium is guaranteed by the result of Rosen [78]. Hence, the ASeqNash is also well defined by this property.

5.2.4 Total Offline Optimum

Finally, the *total offline optimum* is characterized by:

$$\begin{aligned}
\min \quad & C(\boldsymbol{f}) \\
\text{s.t.} \quad & \sum_{a\in\delta^+(v)} f_a^{ij} - \sum_{a\in\delta^-(v)} f_a^{ij} = \gamma_{ij}(v) && \forall\, v \in V,\ ij \in [K_i],\ i \in [n] \\
& f_a^{ij} \ge 0 && \forall\, a \in A,\ ij \in [K_i],\ i \in [n],
\end{aligned} \tag{5.8}$$

where $\gamma_{ij}(v)$ is defined as in (5.1).

For a given sequence σ, we denote by $\textsc{Opt}(\sigma)$ the optimal value of this convex problem.

5.3 Competitive Analysis – The Nonatomic Case

For a solution $\boldsymbol{f}$ produced by an online algorithm Alg for a given sequence of games σ, we denote by $\textsc{Alg}(\sigma) = C(\boldsymbol{f})$ its cost.

In order to derive competitive results for NSeqNash, we use a similar technique as in Chapter 3. We apply the variational inequality 5.5 several times. For this reason, we define the following function.

Definition 5.5
For a given sequence of games σ with n games and a flow $\boldsymbol{f}$ that is produced by NSEQNASH, we define

$$V^i(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i) := \sum_{a\in A} \ell_a\Big(\sum_{k=1}^{i} f_a^k\Big)\,(x_a^i - f_a^i)$$
$$V(\boldsymbol{f},\boldsymbol{x},n) := \sum_{i=1}^{n} V_i(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i),$$

where $\boldsymbol{x}^1,\dots,\boldsymbol{x}^n$ is any feasible flow.

Lemma 5.6. *A feasible flow $\boldsymbol{f}$ for a sequence of games σ that is produced by* NSEQNASH *satisfies:*

$$V(\boldsymbol{f},\boldsymbol{x},n) \geq 0, \ \text{for all feasible flows } \boldsymbol{x} \text{ for } \sigma.$$

Furthermore,

$$V(\boldsymbol{f},\boldsymbol{x},n) = \sum_{a\in A} V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,n),$$

where $V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,n)$ is defined as

$$V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,n) := \sum_{i=1}^{n} \ell_a\Big(\sum_{k=1}^{i} f_a^k\Big)\,(x_a^i - f_a^i).$$

Proof. From Lemma 5.1 we know that $V^i(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i)$ is nonnegative for all $i = 1,\dots,n$. Summing over i proves the first claim. The second claim follows by changing the summation order. □

We use a simple technique to derive upper bounds on the competitive ratio for NSEQNASH. The idea is to add the nonnegative function $V(\boldsymbol{f},\boldsymbol{x},n)$ given in Lemma 5.6 to the cost of the flow $\boldsymbol{f}$ produced by NSEQNASH. We define for every $a \in A$ and nonnegative vectors $\boldsymbol{f}_a,\boldsymbol{x}_a \in \mathbb{R}_+^{\mathcal{K}}$ the following values (we assume by convention $0/0 = 0$):

$$\omega(\ell_a;n,\lambda) := \sup_{\boldsymbol{f}_a,\boldsymbol{x}_a\geq 0} \frac{C_a(f_a) - \lambda\, C_a(x_a) + V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,n)}{C_a(f_a)}. \tag{5.9}$$

For a given class $\mathcal{L}$ of latency functions and a nonnegative real number $\lambda \geq 0$, we further define

$$\omega(\mathcal{L};n,\lambda) := \sup_{\ell_a\in\mathcal{L}} \omega(\ell_a;n,\lambda).$$

We define the following feasible set for the parameter λ.

Definition 5.7 (Feasible Scaling Set)
The feasible scaling set for λ is defined as

$$\Lambda(\mathcal{L},n) := \left\{\lambda \in \mathbb{R}^+ \middle| \big(1-\omega(\mathcal{L};n,\lambda)\big) > 0\right\}.$$

Theorem 5.8. *Consider a sequence σ of n games and latency functions in $\mathcal{L}$. Then, the competitive ratio of* NSeqNash *for the* OnlineNG*is at most*

$$\inf_{\lambda \in \Lambda(\mathcal{L},n)} \left[\lambda \, (1 - \omega(\mathcal{L}; n, \lambda)^{-1}) \right].$$

Proof. Let $\boldsymbol{f}$ be the flow generated by NSeqNash and let $\boldsymbol{x}$ be any feasible flow for a given sequence of games $\sigma = 1, \ldots, n$. Then, we obtain:

$$\begin{aligned} C(\boldsymbol{f}) &\le C(\boldsymbol{f}) + \lambda \, C(\boldsymbol{x}) - \lambda \, C(\boldsymbol{x}) + V(\boldsymbol{f}, \boldsymbol{x}, n) \qquad (5.10)\\ &= \sum_{a \in A} \left[C_a(f_a) + \lambda \, C_a(x_a) - \lambda \, C_a(x_a) + V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, n) \right] \\ &\le \lambda \, C(\boldsymbol{x}) + \omega(\mathcal{L}; n, \lambda) \, C(\boldsymbol{f}). \qquad (5.11) \end{aligned}$$

Here, (5.10) follows from the variational inequality stated in Lemma 5.6. The last inequality (5.11) follows from the definition of $\omega(\mathcal{L}; n, \lambda)$ and since $\lambda \in \Lambda(\mathcal{L}, n)$. Taking $\boldsymbol{x}$ as the optimal offline solution yields the claim. □

Using the notation:

$$\vartheta_a^n(\ell_a, \boldsymbol{f}_a) := \ell_a(f_a) f_a - \sum_{i=1}^{n} \ell_a\Big(\sum_{k=1}^{i} f_a^k \Big) f_a^i.$$

we can simplify the value $\omega(\mathcal{L}; n, \lambda)$.

Lemma 5.9. *The value $\omega(\ell_a; n, \lambda)$ is at most*

$$\sup_{x_a, \boldsymbol{f}_a \ge 0} \frac{\big(\ell_a(f_a) - \lambda \, \ell_a(x_a)\big) x_a + \vartheta_a^n(\ell_a, \boldsymbol{f}_a)}{\ell_a(f_a) f_a}. \qquad (5.12)$$

Proof. First note that

$$\begin{aligned} C_a(f_a) + V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, n) &= \vartheta_a^n(\ell_a, \boldsymbol{f}_a) + \sum_{i=1}^{n} \ell_a\big(\sum_{k=1}^{i} f_a^k \big) x_a^i \\ &\le \vartheta_a^n(\ell_a, \boldsymbol{f}_a) + \ell_a(f_a) \, x_a, \end{aligned}$$

where the last inequality is valid since latency functions are nondecreasing. Then, using

$$\ell_a(f_a) \, x_a - \lambda \, C_a(x_a) = \big(\ell_a(f_a) - \lambda \, \ell_a(x_a)\big) x_a,$$

yields the claim. □

Figure 5.2 illustrates the value $\vartheta_a^n(\ell_a, \boldsymbol{f}_a)$ for $n = 3$.

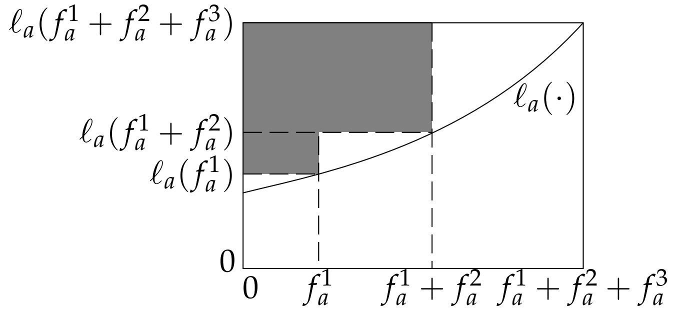

Figure 5.2: Illustration of the value $\vartheta_a^n(\ell_a, f)$ for $n = 3$. The shaded area corresponds to the value $\vartheta_a^n(\ell_a, f)$.

5.3.1 Affine Linear Latency Functions

In the following, we bound the value $\omega(\mathcal{L}; n, \lambda)$ for the class $\mathcal{L}_d$ of polynomials with nonnegative coefficients and degree at most $d \in \mathbb{N}$:

$$\mathcal{L}_d := \{a_d\, x^d + \cdots + a_1\, x + a_0 \;:\; a_s \geq 0, s = 0, \ldots, d\}.$$

We start with the class $\mathcal{L}_1$ but first present some useful prerequisites.

Lemma 5.10. *For parameters $\kappa_1, \kappa_2 > 0$ and any numbers $x, y \geq 0$ the following inequality is valid:*

$$xy \leq \frac{\kappa_1}{2\,\kappa_2} x^2 + \frac{\kappa_2}{2\,\kappa_1} y^2. \tag{5.13}$$

Proof. We use the inequality

$$0 \leq \big(\kappa_1\, x - \kappa_2\, y\big)^2 = \kappa_1^2\, x^2 - 2\,\kappa_1\,\kappa_2\, x\, y + \kappa_2^2\, y^2.$$

Dividing by $2\,\kappa_1\,\kappa_2$ yields the claim. □

Lemma 5.11. *For latency functions in $\mathcal{L}_1$ the value $\omega(\mathcal{L}_1; n, 1)$ is at most $\frac{3n-2}{4n}$.*

Proof. First, by using equation (3.10), we have that

$$\vartheta_a^n(\ell_a, f_a) = q_a\, \tfrac{1}{2} f_a^2 - q_a\, \tfrac{1}{2} \sum_{k=1}^{n} (f_a^k)^2.$$

Then, we obtain

$$\begin{aligned}
\omega(\ell_a; n, 1) &= \sup_{x_a, f_a \geq 0} \frac{q_a\,(f_a - x_a)\, x_a + q_a\, \tfrac{1}{2} f_a^2 - q_a\, \tfrac{1}{2} \sum_{k=1}^{n} (f_a^k)^2}{q_a\, f_a^2 + r_a\, f_a} \\
&\leq \sup_{x_a, f_a \geq 0} \frac{(f_a - x_a)\, x_a + \tfrac{1}{2} f_a^2 - \tfrac{1}{2} \sum_{k=1}^{n} (f_a^k)^2}{f_a^2} && (5.14) \\
&\leq \sup_{x_a, f_a \geq 0} \frac{(f_a - x_a)\, x_a + \frac{n-1}{2n} f_a^2}{f_a^2} && (5.15) \\
&\leq \tfrac{3n-2}{4n}, && (5.16)
\end{aligned}$$

where (5.14) is valid since $r_a \geq 0$. Inequality (5.15) follows from Cauchy-Schwarz inequality and (5.16) follows from Lemma 5.10, where we set $x = f_a, y = x_a, \kappa_1 = 1$, and $\kappa_2 = 2$. □

Equipped with the above lemma, we can prove an upper bound on the competitive ratio of NSeqNash for affine linear latency functions.

Corollary 5.12. *If the latency functions of the* OnlineNG *are in $\mathcal{L}_1$, the online algorithm* NSeqNash *is $\frac{4n}{n+2}$-competitive, where n is the number of games.*

Proof. We bound $\omega(\mathcal{L}_1; n, 1)$ by $\frac{3n-2}{4n}$ using Lemma 5.11. Therefore, choosing $\lambda = 1 \in \Lambda(\mathcal{L}_1, n)$ and applying Theorem 5.8 yields the desired result. □

For $n = 1$, we obtain the bound of $\frac{4}{3}$ for nonatomic network games involving affine linear latency functions presented in Theorem 4.11.

Now, we analyze the case of purely linear latency functions, i.e. the class $\mathcal{M}_1$.

Lemma 5.13. *For latency functions in $\mathcal{M}_1$, we have*

$$\omega(\mathcal{M}_1; n, \lambda) \leq \frac{n + 2\,\lambda\, n - 2\,\lambda}{4\,\lambda\, n}.$$

Proof. The proof proceeds along the line of the proof of the preceeding lemma.

$$\begin{aligned}
\omega(\ell_a; n, \lambda) &\leq \sup_{x_a, f_a \geq 0} \frac{(f_a - \lambda\, x_a)\, x_a + \frac{1}{2} f_a^2 - \frac{1}{2} \sum_{k=1}^{n} (f_a^k)^2}{f_a^2} \\
&\leq \sup_{x_a, f_a \geq 0} \frac{(f_a - \lambda\, x_a)\, x_a + \frac{n-1}{2n} f_a^2}{f_a^2} \\
&\leq \frac{1}{4\,\lambda} + \frac{n-1}{2n}.
\end{aligned}$$

The last inequality follows from Lemma 5.10, where we set $x = x_a, y = f_a, \kappa_1 = \lambda$, and $\kappa_2 = \frac{1}{2}$. Rewriting yields the result. □

Corollary 5.14. *For latency functions in $\mathcal{M}_1$, the online algorithm* NSeqNash *is $\frac{4n^2}{(n+1)^2}$-competitive, where n is the number of games.*

Proof. We bound $\omega(\mathcal{M}_1; n, \lambda)$ from above by $\frac{n+2\,\lambda\, n-2\,\lambda}{4\,\lambda\, n}$. In order to find an optimal λ in Theorem 5.8, we need to ensure that $\lambda \in \Lambda(\mathcal{M}_1, n)$. Hence, we need

$$\frac{n + 2\,\lambda\, n - 2\,\lambda}{4\,\lambda\, n} < 1.$$

This condition leads to $\lambda > \frac{n}{2\,(n+1)}$. Setting $\lambda := \frac{n}{n+1}$ and applying Theorem 5.8 yields

$$C(\boldsymbol{f}) \leq \frac{4\,\lambda^2\, n}{2\,\lambda\, n - n + 2\,\lambda}\, C(\boldsymbol{x}) = \frac{4\, n^2}{(n+1)^2}\, C(\boldsymbol{x}).$$

□

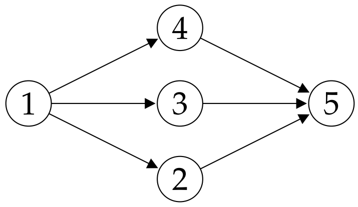

Figure 5.3: Graph construction for the proofs of Proposition 5.16

.

Remark 5.15. The value $\lambda = \frac{n}{n+1}$ solves the following minimization problem with respect to λ:

$$\min_{\lambda \geq \frac{n}{2(n+1)}} \frac{4\lambda^2 n}{2\lambda n - n + 2\lambda}.$$

Interestingly, we get the same upper bound as for the online algorithm Seq within the framework OnlineMCRP for *affine* linear price functions.

5.3.2 Lower Bounds

We start with a result that holds for *any* deterministic online algorithm.

Proposition 5.16. *In case of latency functions in $\mathcal{M}_1$ no deterministic online algorithm for* OnlineNG *is c-competitive for any $c < \frac{4}{3}$.*

Proof. Consider the network displayed in Figure 5.3. Each arc a leaving from node 1 has the same latency function $\ell_a(z) = 3z$. All the other (those leading into node 5) have the latency function $\ell_a(z) = 0$. Let Alg be an arbitrary deterministic online algorithm. We first present Alg commodity 1 with demand 1 that has to be routed from $s_1 = 1$ to $t_1 = 5$. First, assume the algorithm behaves like the NSeqNash. This means that the demand gets evenly divided into three pieces: one third is routed over path $P_1 = (1, 2, 5)$, another over path $P_2 = (1, 3, 5)$, and the later over path $P_3 = (1, 4, 5)$. In this case, we reveal commodity 2 with demand 1 between 1 and 2. For this commodity there exists a unique path. Therefore, Alg yields for this sequence σ the cost:

$$\text{Alg}(\sigma) = \text{NSeqNash}(\sigma) = 2 \cdot 3 \cdot \tfrac{1}{3} \cdot \tfrac{1}{3} + 3 \cdot \left(\tfrac{1}{3} + 1\right)^2 = 6.$$

An optimal offline solution is to route half of commodity 1 over path P_2 and the other half over path P_3 and commodity 2 along its unique path. Therefore,

$$\text{Opt}(\sigma) = 2 \cdot 3 \cdot \tfrac{1}{2} \cdot \tfrac{1}{2} + 3 \cdot 1 \cdot 1 = \tfrac{9}{2}.$$

This leads to

$$\frac{\text{Alg}(\sigma)}{\text{Opt}(\sigma)} = \frac{4}{3}.$$

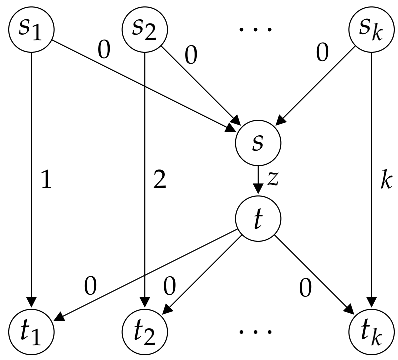

Figure 5.4: Graph construction for the proof of Theorem 5.17.

If Alg does not behave like NSeqNash for the first commodity, Alg has to route more than one third of the demand over path P_1, path P_2, or path P_3. If it is path P_1, then we present commodity 2 as above. If its path P_2, then we reveal a commodity 2 with demand 1 between 1 and 3. Otherwise, we present a commodity 2 with demand 1 between 1 and 4. Let α be the demand greater than one third. In all three cases the cost of Alg for the sequence σ is

$$\text{Alg}(\sigma) \geq 2 \cdot 3 \cdot \left(\tfrac{(1-\alpha)}{2}\right)^2 + 3 \cdot \left(\alpha + 1\right)^2 > 6.$$

since $\alpha > \frac{1}{3}$. The optimal cost stays the same as above. Hence,

$$\frac{\text{Alg}(\sigma)}{\text{Opt}(\sigma)} > \frac{4}{3}.$$

□

For NSeqNash we can further lift the lower bound.

Theorem 5.17. *In case of latency functions in $\mathcal{L}_1$, the online algorithm* NSeqNash *for the* OnlineNG *has a competitive ratio greater than or equal to $\frac{3n-2}{n}$, where n is the number of games.*

Proof. We consider the network presented in Figure 5.4 with the latency functions: $\ell_{(s_i,s)}(z) = 0$, $\ell_{(t,t_i)}(z) = 0$, $\ell_{(s_i,t_i)}(z) = i$, $i = 1, \ldots, k$, and $\ell_{(s,t)}(z) = z$. We consecutively release a sequence of games $(1, \ldots, k)$, where in each game j, there is a single player type $j1$. The demand of player type $j1$ is 1 that has to be routed from s_i to t_i, for $i = 1, \ldots, k$. Due to the choice of the affine terms i, NSeqNash routes for every game the corresponding demand over the arc from s to t. Then we release the $(k+1)$-th game with demand d from s to t. Thus, the total cost for the sequence $\sigma = (1, \ldots, k+1)$ for NSeqNash with the new cost function is given by:

$$\text{NSeqNash}(\sigma) = (k+d)^2.$$

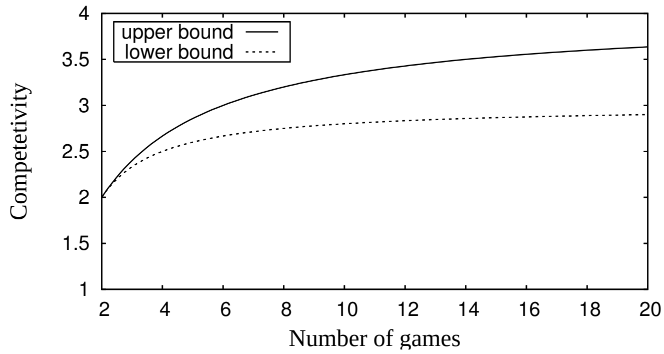

Figure 5.5: Upper bound $\frac{4n}{(n+2)}$ versus lower bound $\frac{3n-2}{n}$ on the competitive ratio of NSeqNash for affine linear latency functions.

The optimal offline algorithm Opt routes the demands of the first k games along the direct arcs from s_i to t_i incurring cost of:

$$\sum_{i=1}^{k} i = \frac{k(k+1)}{2}.$$

The last demand in game $k+1$ is routed from s to t with cost d^2. The total cost for the sequence $\sigma = (1, \ldots, k+1)$ for Opt is given by:

$$\textsc{Opt}(\sigma) = \frac{k(k+1)}{2} + d^2.$$

Replacing $k = n-1$ and setting $d = \frac{n}{2}$ yields

$$\frac{\textsc{NSeqNash}(\sigma)}{\textsc{Opt}(\sigma)} = \frac{2(k+d)^2}{k(k+1)+2d^2} = \frac{3n-2}{n}, \tag{5.17}$$

which proves the theorem. □

Remark 5.18. For $n = 2$, the upper bound given in Corollary 5.12 is tight.

In the following, we present a lower bound for latency functions in $\mathcal{M}_1$.

Corollary 5.19. *For latency functions in $\mathcal{M}_1$, the online algorithm* NSeqNash *for* OnlineNG *has a competitive ratio greater than or equal to* $\frac{33+5\sqrt{33}}{33+\sqrt{33}}$.

Proof. We consider the network presented in Figure 5.4 with modified latency functions: $\ell_{(s_i,s)}(z) = 0$, $\ell_{(t,t_i)}(z) = 0$, $\ell_{(s_i,t_i)}(z) = i\,z$, $i = 1, \ldots, k$, and $\ell_{(s,t)}(z) = z$. We consecutively release a sequence of games $(1, \ldots, k)$, where in each game j, there is a single player type $j1$. The demand of player type $j1$ is 2 that has to be routed from s_i to t_i, for $i = 1, \ldots, k$. Due to the choice of the linear terms $i\,z$, NSeqNash routes for every game the one unit of the demand over the arc from s to t and the other unit along the direct arc from s_i to t_i. To see this, consider the j-th game. Let the flow of player $j1$ along the middle arc be denoted by x. Then, using the characterization of a Nash flow given in (5.4), the nonatomic player $j1$ sends flow x^* along the middle arc according to the solution of the following problem

$$\min_{0 \le x \le 2} \frac{1}{2}\, j\, x^2 + (j-1)\, x + \frac{1}{2}\, j\, (2-x)^2.$$

The solution to this concave program is given by $x^* = 1$, independently of j.

Then, we release the $(k+1)$-th game with demand d from s to t. Thus, the total cost for the sequence $\sigma = (1, \ldots, k+1)$ for NSeqNash is given by:

$$\textsc{NSeqNash}(\sigma) = \sum_{i=1}^{k} i + (k+d)^2 = \frac{k\,(k+1)}{2} + (k+d)^2.$$

The optimal offline algorithm Opt routes the demands of the first k games along the direct arcs from s_i to t_i incurring cost of:

$$\sum_{i=1}^{k} (i \cdot 2) \cdot 2 = 2\,k\,(k+1).$$

The last demand in game $k+1$ is routed from s to t with cost d^2. The total cost for the sequence $\sigma = (1, \ldots, k+1)$ for Opt is given by:

$$\textsc{Opt}(\sigma) = 2\,k\,(k+1) + d^2.$$

Replacing $k = n-1$ and setting $d = \frac{1}{4}\,n + \frac{1}{2} + \frac{1}{4}\sqrt{33\,n^2 - 28\,n + 4}$ yields

$$\frac{\textsc{NSeqNash}(\sigma)}{\textsc{Opt}(\sigma)} \ge \lim_{n\to\infty} Z(n) = \frac{33 + 5\sqrt{33}}{33 + \sqrt{33}} \approx 1.59,$$

where we define

$$Z(n) := \frac{33\,n^2 - 28\,n + 5\,n\sqrt{33\,n^2 - 28\,n + 4} + 4 - 2\sqrt{33\,n^2 - 28\,n + 4}}{33\,n^2 - 28\,n + n\sqrt{33\,n^2 - 28\,n + 4} + 4 + 2\sqrt{33\,n^2 - 28\,n + 4}}.$$

This proves the claim. □

Remark 5.20. The parameter d in the previous proof is the optimal solution to the following maximization problem with optimal value $Z(k+1)$:

$$\max_{d \ge 1} \frac{k\,(k+1) + 2\,(k+d)^2}{4\,k\,(k+1) + 2\,d^2} = Z(k+1).$$

The table below summarizes the main results for (affine) linear latency functions.

Table 5.1: Competitive Ratio for the online algorithm NSeqNash for affine linear latency functions $\ell(x) = a_1\, x + a_0$, $a_0 \geq 0$, $a_1 \geq 0$. The first row shows known results for nonatomic network games. The $\frac{4}{3}$ result is due to Roughgarden and Tardos [84]. UB and LB abbreviates upper bound and lower bound, respectively.

# games		$\ell_a(0) = 0$		$\ell_a(0)$ arbitrary, $\lambda = 1$	
	λ	UB	LB	UB	LB
1	1	1	1	$\frac{4}{3}$	$\frac{4}{3}$
2	$\frac{2}{3}$	$1\,\frac{7}{9}$	$\frac{5+2\sqrt{5}}{5+\sqrt{5}}$	2	2
3	$\frac{3}{4}$	$2\,\frac{1}{4}$	$\frac{217+13\sqrt{217}}{217+5\sqrt{217}}$	$2\,\frac{2}{5}$	$2\,\frac{1}{3}$
	.	.	.	.	
	.	.	.		
n	$\frac{n}{n+1}$	$\frac{4\,n^2}{(n+1)^2}$	$Z(n)$	$\frac{4\,n}{n+2}$	$\frac{3n}{n-2}$
∞	1	4	$\frac{33+5\sqrt{33}}{33+\sqrt{33}}$	4	3

5.3.3 Polynomial Latency Functions

In this section, we investigate the case, where we allow for polynomial latency functions in $\mathcal{L}_d$.

We start with a useful observation.

Lemma 5.21. *For latency functions in $\mathcal{L}_d$, we can bound $\sup\limits_{f_a \geq 0} \vartheta_a^n(\ell_a, f_a)$ as follows:*

$$\sup_{f_a \geq 0} \vartheta_a^n(\ell_a, f_a) \leq \sup_{f_a \geq 0} \vartheta_a^n(\ell_a, f_a) \leq \frac{d}{d+1}\, \ell_a(f_a)\, f_a,$$

where $\vartheta_a^n(\ell_a, f_a) := \lim\limits_{n \to \infty} \vartheta_a^n(\ell_a, f_a)$.

Proof. Recall the definition of $\vartheta_a^n(\ell_a, f_a)$:

$$\vartheta_a^n(\ell_a, f_a) := \ell_a(f_a)\, f_a - \sum_{i=1}^{n} \ell_a\Big(\sum_{k=1}^{i} f_a^k\Big)\, f_a^i.$$

Since polynomials with nonnegative coefficients are nondecreasing functions, the following inequalities hold

$$\inf_{f_a \geq 0} \Big[\sum_{i=1}^{n} \ell_a\Big(\sum_{k=1}^{i} f_a^k\Big)\, f_a^i\Big] \geq \inf_{f_a \geq 0} \Big[\sum_{i=1}^{\infty} \ell_a\Big(\sum_{k=1}^{i} f_a^k\Big)\, f_a^i\Big] \geq \int_0^{f_a} l(z)\, dz.$$

Hence, we have

$$\sup_{f_a \geq 0} \vartheta_a^n(\ell_a, \boldsymbol{f}_a) \leq \sup_{\boldsymbol{f}_a \geq 0} \vartheta_a^n(\ell_a, \boldsymbol{f}_a) \leq \ell_a(f_a)\, f_a - \int_0^{f_a} l(z)\, dz. \tag{5.18}$$

Let $\ell_a(z) = \sum_{i=0}^{d} a_i\, z^i$ be a polynomial of degree $d \geq 1$. Then, it follows that

$$\begin{aligned}
\ell_a(f_a)\, f_a - \int_0^{f_a} l(z)\, dz &= \sum_{i=0}^{d} a_i\, (f_a)^{i+1} - \sum_{i=0}^{d} \big(\frac{1}{i+1}\big)\, a_i\, (f_a)^{i+1} \\
&= \sum_{i=0}^{d} \big(\frac{i}{i+1}\big)\, a_i\, (f_a)^{i+1} \\
&\leq \frac{d}{d+1} \sum_{i=0}^{d} a_i\, (f_a)^{i+1} \\
&= \frac{d}{d+1}\, \ell_a(f_a)\, f_a.
\end{aligned}$$

Using inequality (5.18), the claim is proven. □

Lemma 5.22. *If the latency functions of the* OnlineNG *are in $\mathcal{L}_d$, $d \geq 1$ and $\lambda \geq 1$, then, the value $\omega(\ell_a, n; \lambda)$ is at most*

$$\max_{0 \leq \mu} \big[\mu - \lambda\, \mu^{d+1}\big] + \frac{d}{d+1}.$$

Proof. By Lemma 5.12, we have

$$\omega(\ell_a, n; \lambda) \leq \sup_{\boldsymbol{f}_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a + \vartheta_a^n(\ell_a, \boldsymbol{f}_a)}{\ell_a(f_a)\, f_a}.$$

Then, using Lemma 5.21 we have that

$$\begin{aligned}
\omega(\ell_a, n; \lambda) &\leq \sup_{f_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a + \frac{d}{d+1}\, \ell_a(f_a)\, f_a}{\ell_a(f_a)\, f_a} \\
&= \sup_{f_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(x_a)\big)\, x_a}{\ell_a(f_a)\, f_a} + \frac{d}{d+1}.
\end{aligned}$$

Defining $\mu := \frac{x_a}{f_a}$ for $f_a > 0$ and zero otherwise, we have to solve

$$\max_{0 \leq \mu} \frac{\big(\ell_a(f_a) - \lambda\, \ell_a(\mu\, f_a)\big)\, \mu\, f_a}{\ell_a(f_a)\, f_a}$$

to bound $\omega(\ell_a, n; \lambda)$ from above. The rest of the proof follows from the proof of Lemma 3.22 in Chapter 3. □

Proposition 5.23. *For latency functions in $\mathcal{L}_d$ and $\lambda := (d+1)^{(d-1)} \geq 1$, the value $\omega(\mathcal{L}; n, \lambda)$ is at most $\frac{d^2+2d}{(d+1)^2}$.*

Proof. We start with Lemma 5.22.

$$\omega(\mathcal{L}_d; n, \lambda) \leq \max_{0 \leq \mu \leq 1} \mu - \lambda\, \mu^{d+1} + \frac{d}{d+1} = \max_{0 \leq \mu \leq 1} \mu - (d+1)^{(d-1)}\, \mu^{d+1} + \frac{d}{d+1}.$$

The unique solution is given by $\mu^* = \frac{1}{d+1}$. Evaluating the objective proves the claim:

$$\omega(\ell_a, n; \lambda) \leq \frac{1}{d+1} - (d+1)^{(d-1)}\, (\frac{1}{d+1})^{d+1} + \frac{d}{d+1} = \frac{d^2+2d}{(d+1)^2}.$$

□

With this lemma we can prove a constant factor bound on the competitive ratio that depends on the degree d of the considered polynomials.

Theorem 5.24. *Consider the* OnlineNG *with latency functions in $\mathcal{L}_d$. Then, the competitive ratio of the online algorithm* NSeqNash *is at most $(d+1)^{d+1}$.*

Proof. Let the flow $\boldsymbol{f}$ be produced by the online algorithm NSeqNash and let $\boldsymbol{x}$ be an arbitrary feasible flow for the OnlineNG. We define $\lambda := (d+1)^{(d-1)}$ and apply Proposition 5.23, which yields $\omega(\mathcal{L}_d; n, \lambda) \leq \frac{d^2+2d}{(d+1)^2}$. In order to apply Theorem 5.8, we have to verify that $\lambda \in \Lambda(\mathcal{L}_d, n)$. What remains to be shown is that

$$1 - \frac{d^2+2d}{(d+1)^2} > 0$$

holds. This inequality is equivalent to

$$\frac{1}{d+1} > 0,$$

which is trivially true. Then, applying Theorem 5.8 yields

$$C(\boldsymbol{f}) \leq \frac{(d+1)^{d-1}}{\left(1 - \frac{d^2+2d}{(d+1)^2}\right)}\, C(\boldsymbol{x}) = (d+1)^{d+1}\, C(\boldsymbol{x}).$$

Taking $\boldsymbol{x}$ as the optimal offline solution proves the claim. □

By optimizing over $\lambda \in \Lambda(\mathcal{L}_d, n)$, we get the following bounds for polynomial latency functions as shown in Table 5.2.

Table 5.2: Competitive ratio of NSeqNash for different polynomial latency functions. Coefficients a_i are assumed to be nonnegative.

Set $\mathcal{L}$ of latency functions	Example	$\omega(\mathcal{L};\infty,\lambda)$	λ	UB	LB
linear functions	$a_1 x + a_0$	$\frac{3}{4}$	1	4	3
quadratic	$\sum_{i=0}^{2} a_i x^i$	0.93	2.18	19.6	7.5
cubic	$\sum_{i=0}^{3} a_i x^i$	$\frac{15}{16}$	64	256	17.32
.	.	.	.	.	.
degree d	$\sum_{i=0}^{d} a_i x^i$	$\frac{d^2+2d}{(d+1)^2}$	$(d+1)^{(d-1)}$	$(d+1)^{d+1}$	$\frac{d+1}{d+2}2^{d+1}$

5.3.4 Lower Bounds for Polynomial Latency Functions

Consider the network presented in Figure 5.4 with the following latency functions: $\ell_{(s_i,s)}(z) = 0$, $\ell_{(t,t_i)}(z) = 0$, $\ell_{(s_i,t_i)}(z) = i^d$, $i = 1,\ldots,k$, and $\ell_{(s,t)}(z) = z^d, d \in \mathbb{N}$. We consecutively release games with a single player type $i1$, where a demand of size 1 has to be routed from s_i to t_i, for $i = 1,\ldots,k$. Due to the choice of the affine terms i^d, NSeqNash routes every demand over the arc from s to t. Then we release the $(k+1)$-th game with demand x from s to t. The total cost for the flow generated by NSeqNash is given by:

$$\text{NSeqNash}(\sigma) = \big(k+x\big)^{d+1}.$$

The optimal offline algorithm Opt routes the demands of the first k games along the direct arcs from s_i to t_i. The last demand is routed from s to t. The total cost for Opt is then given by:

$$\text{Opt}(\sigma) = \sum_{i=1}^{k} i^d + x^{d+1}.$$

From Lemma 3.25, we know that the d-th power of the sum of numbers from 1 to k is a polynomial in k given by:

$$\sum_{i=1}^{k} i^d = \frac{1}{d+1} \sum_{j=0}^{d+1} \binom{d+1}{j} B_j\, k^{d+1-j},$$

where B_j are the Bernoulli numbers.

Theorem 5.25. *In case of latency functions in $\mathcal{L}_d$, the online algorithm* NSeqNash *for* OnlineNG *has a competitive ratio greater than or equal to $\frac{d+1}{d+2}\,2^{d+1}$.*

Proof. We have to show that the competitive ratio fulfills:

$$\frac{\text{NSeqNash}_d(\sigma)}{\text{Opt}_d(\sigma)} \geq \frac{d+1}{d+2}\,2^{d+1}.$$

We follow the construction of the above discussion,

$$\frac{\text{NSeqNash}_d(\sigma)}{\text{Opt}_d(\sigma)} \geq \lim_{k\to\infty} \frac{\left(k+x\right)^{d+1}}{\sum\limits_{i=1}^{k} i^d + x^{d+1}}.$$

We set $x = k$ which yields:

$$\begin{aligned}\frac{\text{NSeqNash}_d(\sigma)}{\text{Opt}_d(\sigma)} &\geq \lim_{k\to\infty} \frac{\left(2k\right)^{d+1}}{\sum\limits_{i=1}^{k} i^d + k^{d+1}} \\ &= \lim_{k\to\infty} \frac{\left(2k\right)^{d+1}}{\frac{1}{d+1} k^{d+1} + k^{d+1} + \sum\limits_{j=1}^{d+1} \binom{d+1}{j} B_j\, k^{d+1-j}} = \frac{d+1}{d+2} 2^{d+1},\end{aligned}$$

where the equality follows from Lemma 3.25 and the fact that $B_0 = 1$.

□

Note that the derived lower bounds are larger than the lower bounds obtained for the greedy online algorithm Seq for the OnlineMCRP in Chapter 3.

5.3.5 Parallel Networks

For graphs that consist of two nodes and parallel arcs, we can show that NSeqNash performs not worse than a Nash flow for the entire game sequence that is played in parallel. In other words, for a given sequence of games, we compare the cost of NSeqNash to the cost of a Nash flow of a complementary game, where all players of the entire game sequence route their demands simultaneously.

Definition 5.26
For a given instance of the OnlineNG involving a sequence of games σ, we define the *complementary game* $\bar{\sigma}$ as a single game that contains all players of the sequence σ simultaneously.

Recall from the Wardrop condition (4.2) that a flow f is at Nash equilibrium if and only if the following condition is satisfied:

Lemma 5.27. *A feasible flow f for the game $\bar{\sigma}$ is a Nash equilibrium if and only if:*

$$\ell_a(f_a) \leq \ell_{\hat{a}}(f_a), \text{ for all arcs } a, \hat{a} \in A \text{ such that } f_a > 0. \tag{5.19}$$

Note that for nonatomic network games, Nash equilibria and Wardrop equilibria are the same. A similar condition holds for the flow that is produced by NSeqNash.

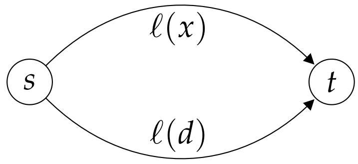

Figure 5.6: Graph construction for a matching lower bound for Theorem 5.29.

Lemma 5.28. *A feasible flow $\boldsymbol{f}$ for the sequence of games σ is produced by* NSeqNash *if and only if for all $k \in [n]$:*

$$\ell_a(\sum_{i=1}^{k} f_a^i) \leq \ell_{\hat{a}}(\sum_{i=1}^{k} f_a^i), \text{ for all edges } a, \hat{a} \in A, \text{ such that } f_a^k > 0. \tag{5.20}$$

Theorem 5.29. *Let $D = (V, A)$ with $V = \{s, t\}$ and A a set of edges from s to t. We are given a sequence of games $\sigma = 1, \ldots, n$. Let $\boldsymbol{f}$ be a flow produced by* NSeqNash *for the* OnlineNG *with a single nonatomic player routing d_i from s to t in every game $i \in [n]$. Let $\boldsymbol{f}^*$ be a flow at Nash equilibrium for the corresponding game $\bar{\sigma}$ with a single player routing $\sum_{i=1}^{n} d_i$ from s to t. Then, $C(\boldsymbol{f}) = C(\boldsymbol{f}^*)$.*

Proof. We prove that the flow $\boldsymbol{f}$ satisfies all conditions of Lemma 5.27 for the game $\bar{\sigma}$. By the uniqueness of the cost of a Nash equilibrium the claim is proven.

The latency of the flow $\boldsymbol{f}$ on edge a is equal $\ell_a(f_a)$. By contradiction assume that there exist edges $a, \hat{a} \in A$ with

$$\ell_a(f_a) > \ell_{\hat{a}}(f_{\hat{a}}), \text{ with } f_a > 0.$$

Let $k \in [n]$ be the largest index with $f_a^k > 0$. The existence of such an index k is granted since $f_a = \sum_{i=1}^{n} f_a^i > 0$ is assumed. As in games $k+1, \ldots, n$, the edge a is not used any more, we have that $\ell_a(f_a) = \ell_a\big(\sum_{i=1}^{k} f_a^i\big)$. Using the assumption that latency functions are nondecreasing it follows that $\ell_{\hat{a}}(f_{\hat{a}}) \geq \ell_{\hat{a}}\big(\sum_{i=1}^{k} f_{\hat{a}}^i\big)$. By Lemma 5.28 for game k, we have $\ell_{\hat{a}}\big(\sum_{i=1}^{k} f_{\hat{a}}^i\big) \geq \ell_a\big(\sum_{i=1}^{k} f_a^i\big)$, thus

$$\ell_{\hat{a}}(f_a) \geq \ell_{\hat{a}}\Big(\sum_{i=1}^{k} f_{\hat{a}}^i\Big) \geq \ell_a\Big(\sum_{i=1}^{k} f_a^i\Big) = \ell_a(f_a),$$

a contradiction. □

A trivial example showing that the above upper bound is tight is to consider a sequence σ that only contains a single game. In this case, it is well known that for classes $\mathcal{L}$, which contain constant terms, the anarchy value $\alpha(\mathcal{L})$ is tight. Matching lower bounds can be derived via Pigou instances, as shown in Figure 5.6. Based on an example given in Correa, Schulz, and Stier-Moses [24], we now show that the upper bound on the competitive ratio of NSeqNash in Theorem 5.29 is tight for an *arbitrary* number of games.

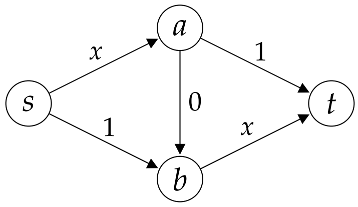

Figure 5.7: Braess Graph.

Consider the Pigou instance in Figure 5.6, see Pigou [75]. Assume that the sequence σ contains n games, where in each game we are given a demand d_i, $i = 1, \ldots, n$. We denote by $d = \sum_{i=1}^{n} d_i$ the aggregated demand. We are given a variable latency functions $\ell(x)$ and a constant latency function $\ell(d)$. The online algorithm NSEQNASH routes all demands along the cheaper upper arc. Hence, after routing the last demand d_n, the latency of the resulting flow f is equal on both arcs. Thus, the cost of the flow is given by $C(f) = d\,\ell(d)$. The system optimum can be evaluated as follows:

$$C(x^*) = \min_{0 \le x \le d} \{x\,\ell(x) + \ell(d)(d - x)\} = d\,\ell(d) - \max_{0 \le x \le d} \{x\,(\ell(d) - \ell(x))\,x\}.$$

Evaluating the ratio between the cost of a flow at Nash equilibrium and the optimal cost yields

$$\begin{aligned} \frac{C(f)}{C(x^*)} &= \left(1 - \frac{\max_{0 \le x \le d}\{x\,(\ell(d) - \ell(x))\,x\}}{d\,\ell(d)}\right)^{-1} \\ &= (1 - \omega(\mathcal{L}, 1))^{-1} = (1 - \beta(\mathcal{L}))^{-1} = \alpha(\mathcal{L}). \end{aligned}$$

Since Theorem 5.29 ensures that the cost of the flow $\boldsymbol{f}$ is upper bounded by the price of anarchy $\alpha(\mathcal{L})$ for the corresponding game $\bar{\sigma}$, these upper bounds are tight by definition. The intuition of the above proof fails, however, for general networks with a single source and a single destination. To see this, we present an instance, where the cost of a flow $\boldsymbol{f}$ produced by NSEQNASH is larger than that of the corresponding Nash flow $\boldsymbol{f}^*$ for the game $\bar{\sigma}$.

Example 5.30. Consider the graph of Braess's paradox in Figure 5.7 and two games that are released consecutively. Each game has a single nonatomic player routing one unit $d_1 = 1$, $d_2 = 1$ from s to t. The path system $\mathcal{P}_1$ for the first player contains $P_1 = (s, a, t)$, $P_2 = (s, a, b, t)$, $P_3 = (s, b, t)$. A flow that is at Nash equilibrium for the first game routes 1 unit of flow on P_2, having path latency $\ell_1(f^1) = 2$. In the second game, we route $\frac{1}{2}$ unit on P_1 and $\frac{1}{2}$ on P_3, both having path latency $\ell_2(\boldsymbol{f}) = 2.5$. Now $\ell^2_{P_2}(\boldsymbol{f}) = 3$. Thus, the total cost is $C(\boldsymbol{f}) = 1 \times 2.5 + 1 \times 3 = 5.5$. However, for the game $\bar{\sigma}$ we route 2 units of flow from s to t. Then, a flow $\boldsymbol{f}^*$ at Nash equilibrium routes one unit along paths P_1 and P_3. The path latencies are $\ell_{P_1}(\boldsymbol{f}) = \ell_{P_2}(\boldsymbol{f}) = 2$, thus the total cost is $C(\boldsymbol{f}^*) = 2 \times 2 = 4$.

This example shows that we can derive a lower bound on the competitive ratio of NSeqNash in this setting evaluating to $5.5/4 > 1$. We do not know if it is possible to improve the upper bound of NSeqNash for the *s*-*t* setting.

5.4 Competitive Analysis – The Atomic Case

In this section, we study the online algorithm ASeqNash, which is the atomic counterpart of NSeqNash. The only difference is, that commodities of game i are controlled and coordinated by the corresponding players. Our used techniques follow along similar lines of the previous sections. Before we state the main result, we need some useful prerequisites.

Definition 5.31
For a given sequence of games σ and a flow $\boldsymbol{f}$ that is produced by ASeqNash, we define

$$\begin{aligned} V^{ij}(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i) &:= \sum_{a\in A}\Big(\ell_a\big(\sum_{k=1}^{i} f_a^k\big) + \ell'_a\big(\sum_{k=1}^{i} f_a^k\big) f_a^{ij}\Big)\,(x_a^{ij} - f_a^{ij}),\\ V^{i}(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i,\mathcal{K}_i) &:= \sum_{ij\in[K_i]} V^{ij}(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i),\\ V(\boldsymbol{f},\boldsymbol{x},\mathcal{K},n) &:= \sum_{i=1}^{n} V^{i}(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i,\mathcal{K}_i), \end{aligned} \tag{5.21}$$

where $\boldsymbol{x}^1,\dots,\boldsymbol{x}^n \in \mathcal{F}_n$ is any feasible flow.

Lemma 5.32. *A feasible flow $\boldsymbol{f}$ for a sequence of games σ that is produced by* ASeqNash *satisfies:*

$$V(\boldsymbol{f},\boldsymbol{x},\mathcal{K},n) \geq 0, \ \textit{for all feasible flows } \boldsymbol{x} \textit{ for } \sigma. \tag{5.22}$$

Furthermore,

$$V(\boldsymbol{f},\boldsymbol{x},\mathcal{K},n) = \sum_{a\in A} V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,\mathcal{K},n),$$

where $V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,\mathcal{K},n)$ is defined as

$$V_a(\boldsymbol{f}_a,\boldsymbol{x}_a,\mathcal{K},n) := \sum_{i=1}^{n}\sum_{ij\in[K_i]}\Big(\ell_a\big(\sum_{k=1}^{i} f_a^k\big) + \ell'_a\big(\sum_{k=1}^{i} f_a^k\big) f_a^{ij}\Big)\,(x_a^{ij} - f_a^{ij}).$$

Proof. From Lemma 5.3 we know that $V^{ij}(\boldsymbol{f}^1,\dots,\boldsymbol{f}^i,\boldsymbol{x}^i)$ is nonnegative for all $ij \in [\mathcal{K}_i]$ and $i = 1,\dots,n$.. Summing over $ij \in [\mathcal{K}_i]$ and $i = 1,\dots,n$ proves the first claim. The second claim follows by changing the summation order. □

We define for every $a \in A$, for any nonnegative vectors $\boldsymbol{f}_a, \boldsymbol{x}_a \in \mathbb{R}_+^{\mathcal{K}}$ the following values (we assume by convention $0/0 = 0$):

$$\omega(\ell_a,n,\mathcal{K};\lambda) := \sup_{\boldsymbol{f}_a,\boldsymbol{x}_a\geq 0} \frac{C_a(f_a) - \lambda\, C_a(x_a) + V_a(\boldsymbol{f}_a,\boldsymbol{x}_a)}{C_a(f_a)}, \tag{5.23}$$

where the notation $\boldsymbol{f}_a, \boldsymbol{x}_a \geq 0$ is defined in (4.20).

For a given class $\mathcal{L}$ of latency functions and a nonnegative real number $\lambda \geq 0$, we further define

$$\omega(\mathcal{L}; n, \mathcal{K}, \lambda) := \sup_{\ell_a \in \mathcal{L}} \omega(\ell_a, n, \mathcal{K}; \lambda).$$

We define the following feasible set for the parameter λ.

Definition 5.33 (Feasible Scaling Set)
The feasible scaling set for λ is defined as

$$\Lambda(\mathcal{L}, n, \mathcal{K}) := \left\{\lambda \in \mathbb{R}^+ | \left(1 - \omega(\mathcal{L}; n, \mathcal{K}, \lambda)\right) > 0\right\}.$$

Theorem 5.34. *Consider an instance of the* OnlineNG *involving a sequence of n games with $\mathcal{K}$ players and latency functions in $\mathcal{L}$. Then, the competitive ratio of* ASeqNash *is at most*

$$\inf_{\lambda \in \Lambda(\mathcal{L}, n, \mathcal{K})} \left[\lambda \left(1 - \omega(\mathcal{L}; n, \mathcal{K}, \lambda)\right)^{-1}\right].$$

Proof. Let $\boldsymbol{f}$ be the flow generated by ASeqNash, and $\boldsymbol{x}$ be any feasible flow for a given sequence of games $\sigma = (1, \ldots, n)$.

$$\begin{aligned} C(\boldsymbol{f}) &\leq \sum_{a \in A} \left[C_a(f_a) + V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, \mathcal{K}, n)\right] &(5.24)\\ &= \sum_{a \in A} \left[C_a(f_a) + \lambda\, C_a(x_a) - \lambda\, C_a(x_a) + V_a(\boldsymbol{f}_a, \boldsymbol{x}_a, \mathcal{K}, n)\right] \\ &\leq \lambda\, C(\boldsymbol{x}) + \omega(\mathcal{L}; n, \mathcal{K}, \lambda)\, C(\boldsymbol{f}). &(5.25)\end{aligned}$$

Here, (5.24) follows from the variational inequality stated in Lemma 5.32. The last inequality (5.25) follows from the definition of $\omega(\mathcal{L}; n, \mathcal{K}, \lambda)$. □

With this result, we can investigate the above infimum expression for different classes of latency functions. The most important and natural functions are polynomials in $\mathcal{L}_d$. We will, however, pay increased attention to linear and affine liner latency functions in the following.

Using the notation:

$$\theta_a^i := \sum_{ij \in [K_i]} (f_a^{ij} x_a^{ij} - f_a^{ij} f_a^{ij}),$$

we can simplify the value $\omega(\mathcal{L}; n, \mathcal{K}, \lambda)$.

Lemma 5.35. *The value $\omega(\ell_a, n, \mathcal{K}; \lambda)$ is at most*

$$\sup_{x_a, \boldsymbol{f}_a \geq 0} \frac{\left(\ell_a(f_a) - \lambda\, \ell_a(x_a)\right) x_a + \vartheta_a^n(\ell_a, \boldsymbol{f}_a) + \sum_{i=1}^{n} \ell_a'(\sum_{k=1}^{i} f_a^k)\, \theta_a^i}{\ell_a(f_a) f_a}. \tag{5.26}$$

Proof. First note that

$$C_a(f_a) + V_a(\boldsymbol{f}_a, \boldsymbol{x}_a) = \vartheta_a^n(\ell_a, \boldsymbol{f}_a) + \sum_{i=1}^{n} \big[\ell_a'(\sum_{k=1}^{i} f_a^k)\,\theta_a^i + \ell_a(\sum_{k=1}^{i} f_a^k) x_a^i\big]$$
$$\leq \vartheta_a^n(\ell_a, \boldsymbol{f}_a) + \sum_{i=1}^{n} \ell_a'(\sum_{k=1}^{i} f_a^k)\,\theta_a^i + \ell_a(f_a)\,x_a,$$

where the last inequality is valid since latency functions are nondecreasing. Then, using

$$\ell_a(f_a)\,x_a - \lambda\,C_a(x_a) = \big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)x_a,$$

yields the claim. □

Note that for $\lambda = 1$ and $n = 1$ the value $\omega(\ell_a, 1, \mathcal{K}; 1)$ is equal to the value $\beta^K(\ell_a)$ defined by Cominetti, Correa, and Stier-Moses in [23]. For $n > 1$, that is, the sequence σ of games contains more than one game, the main difference between $\beta^K(\ell_a)$ and $\omega(\ell_a, n, \mathcal{K}; \lambda)$ are the values $\lambda \geq 0$ and $\vartheta_a^n(\ell_a, \boldsymbol{f}_a)$. The value $\vartheta_a^n(\ell_a, \boldsymbol{f}_a)$ penalizes the efficiency of ASeqNash for multiple games. The value λ admits a further degree of freedom to strengthen the analysis.

5.4.1 Affine Linear Latency Functions

We analyze in the following the value $\omega(\mathcal{L}_1; n, \mathcal{K}, 1)$ for affine linear latency functions in $\mathcal{L}_1$.

Lemma 5.36. *For latency functions in $\mathcal{L}_1$ and $\lambda \geq 1$ the value $\omega(\mathcal{L}_1; n, \mathcal{K}, \lambda)$ is less than or equal to $\frac{4(\mathcal{K}-1)}{5\mathcal{K}+1}$.*

Proof. We start with equation (5.26) for latency functions in $\mathcal{L}_1$. The value $\omega(\ell_a, n, \mathcal{K}; \lambda)$ is at most:

$$\sup_{x_a, \boldsymbol{f}_a \geq 0} \frac{q_a\big(f_a - \lambda\,x_a\big)x_a + q_a\,(f_a)^2 - q_a\,\sum_{i=1}^{n}(\sum_{k=1}^{i} f_a^k)\,f_a^i + q_a\,\sum_{i=1}^{n}\theta_a^i}{q_a(f_a)^2 + r_a\,f_a}$$

$$\leq \sup_{x_a, \boldsymbol{f}_a \geq 0} \frac{\big(f_a - \lambda\,x_a\big)x_a + \frac{1}{2}\,(f_a)^2 - \frac{1}{2}\,\sum_{i=1}^{n}(f_a^i)^2 + \sum_{i=1}^{n}\theta_a^i}{(f_a)^2} \tag{5.27}$$

$$\leq \sup_{x_a, \boldsymbol{f}_a \geq 0} \frac{\big(f_a - \lambda\,x_a\big)x_a + \frac{1}{2}\,(f_a)^2 - \frac{1}{2}\,\sum_{ij \in \mathcal{K}}(f_a^{ij})^2 + \sum_{i=1}^{n}\theta_a^i}{(f_a)^2} \tag{5.28}$$

$$= \sup_{x_a, \boldsymbol{f}_a \geq 0} \frac{\big(f_a - \lambda\,x_a\big)x_a + \frac{1}{2}\,(f_a)^2 + \sum_{ij \in \mathcal{K}}\big(f_a^{ij}\,x_a^{ij} - \frac{3}{2}\,(f_a^{ij})^2\big)}{(f_a)^2},$$

where (5.27) follows from (3.10) and $r_a \geq 0$. Note that to obtain the first inequality we have used that $r_a - \lambda\, r_a \leq 0$ since $\lambda \geq 1$. Inequality (5.28) is valid since the sum of powers is less than the power of the sum. Without loss of generality, we can assume that $f_a^1 := \max_{ij \in [\mathcal{K}]} f_a^{ij}$. Since the individual components x_a^{ij} appear linearly in the expression $f_a^{ij} x_a^{ij}$, we can set $\boldsymbol{x}_a := (x_a^1, 0, \ldots, 0)$ to bound the above expression from above. Thus, we have:

$$\omega(\ell_a, n, \mathcal{K}; \lambda) \leq \sup_{0 \leq f_a^1 \leq f_a, x_a^1 \geq 0} \frac{f_a\, x_a^1 - \lambda\,(x_a^1)^2 + \frac{1}{2}\,(f_a)^2 + f_a^1\, x_a^1 - \sum_{ij \in \mathcal{K}} \frac{3}{2}\,(f_a^{ij})^2}{(f_a)^2}.$$

Because of symmetry in the last sum of the numerator, we can set $f_a^{ij} = \frac{f_a}{\mathcal{K}-1}$.

$$\omega(\ell_a, n, \mathcal{K}) \leq \sup_{\substack{\frac{f_a}{\mathcal{K}} \leq f_a^1 \leq f_a \\ x_a^1 \geq 0}} \frac{f_a x_a^1 - \lambda\,(x_a^1)^2 + \frac{1}{2}\,(f_a)^2 + f_a^1\, x_a^1 - \frac{3}{2}(f_a^1)^2 - \frac{3(f_a - f_a^1)^2}{2(\mathcal{K}-1)}}{(f_a)^2}.$$

For any choice of f_a, f_a^1, the optimal value for x_a^1 is exactly $x_a^1 = \frac{f_a + f_a^1}{2\lambda}$. Inserting the value yields:

$$\omega(\ell_a, n, \mathcal{K}; \lambda) \leq \sup_{\frac{f_a}{\mathcal{K}} \leq f_a^1 \leq f_a} \frac{(\frac{1}{2} + \frac{1}{4\lambda})(f_a)^2 + (\frac{1}{4\lambda} - \frac{3}{2})\,(f_a^1)^2 + \frac{1}{2} f_a^1\, f_a - \frac{3(f_a - f_a^1)^2}{2(\mathcal{K}-1)}}{(f_a)^2}.$$

We replace $f_a^1 = \mu\, f_a$ with $\mu \in [\frac{1}{\mathcal{K}}, 1]$ and solve:

$$\omega(\ell_a, n, \mathcal{K}; \lambda) \leq \max_{\mu \in [\frac{1}{\mathcal{K}}, 1]} (\tfrac{1}{2} + \tfrac{1}{4\lambda}) + (\tfrac{1}{4\lambda} - \tfrac{3}{2})\,\mu^2 + \tfrac{1}{2}\,\mu - \tfrac{3(1-\mu)^2}{2(\mathcal{K}-1)}. \tag{5.29}$$

Now we set $\lambda := 1$. Then, the optimal choice is $\mu = \frac{(\mathcal{K}+5)}{5\mathcal{K}+1}$. This leads to:

$$\omega(\mathcal{L}_1; n, \mathcal{K}, 1) \leq \tfrac{4(\mathcal{K}-1)}{5\mathcal{K}+1}.$$

□

Applying Theorem 5.34 with the above value leads to the following result.

Corollary 5.37. *If the latency functions of the* OnlineNG *are in $\mathcal{L}_1$, the online algorithm* ASeqNash *is $\frac{5\mathcal{K}+1}{\mathcal{K}+5}$-competitive, where $\mathcal{K}$ is the total number of players.*

Proof. Applying Theorem 5.34 with $\lambda = 1$ yields:

$$C(\boldsymbol{f}) \leq \frac{1}{1 - \frac{4(\mathcal{K}-1)}{5\mathcal{K}+1}}\, C(\boldsymbol{x}) = \tfrac{5\mathcal{K}+1}{\mathcal{K}+5}\, C(\boldsymbol{x}).$$

This proves the corollary. □

Corollary 5.37 gives abound that only depends on the total number of players in the sequence σ of games. This bound states that ASEQNASH is asymptotically 5-competitive for online atomic network games.

If we optimize over the parameter $\lambda \in \Lambda(\mathcal{L}_1, n, \mathcal{K})$, we can derive even better bounds. For ease of presentation we focus on the asymptotic bound, that is, we consider the case $\mathcal{K} \to \infty$.

Corollary 5.38. *If the latency functions of the* ONLINENG *are in $\mathcal{L}_1$, the online algorithm* ASEQNASH *is* 4.92*-competitive.*

Proof. We start with bounding $\omega(\mathcal{L}_1; \infty, \infty, \lambda)$ using (5.29):

$$\omega(\ell_a, \infty, \infty; \lambda) \leq \max_{\mu \in [0,1]} (\tfrac{1}{2} + \tfrac{1}{4\lambda}) + (\tfrac{1}{4\lambda} - \tfrac{3}{2})\, \mu^2 + \tfrac{1}{2}\, \mu.$$

Then, it follows that

$$\mu^* = \frac{1}{6\lambda - 1},$$

and

$$\omega(\mathcal{L}; \infty, \infty, 1) \leq \frac{4\lambda + 13\lambda^2 - 1}{4\lambda(6\lambda - 1)}.$$

Note, that we still have $\omega(\mathcal{L}; \infty, \infty, 1) \leq \frac{1}{5}$ for $\lambda = 1$. Applying Theorem 5.34 with $\lambda = 1.13$ yields the claim. □

In the following, we derive a bound that depends on the number of games.

Corollary 5.39. *If the latency functions of the* ONLINENG *are in $\mathcal{L}_1$, the online algorithm* ASEQNASH *is* $\frac{2(3\mathcal{K}+1)n}{n\mathcal{K}+3n+3\mathcal{K}+1}$*-competitive, where n is the number of games and $\mathcal{K}$ is the total number of players.*

Proof. We start with equation (5.27) in Lemma 5.36 to derive another bound on the value $\omega(\mathcal{L}_1; n, \mathcal{K}, \lambda)$.

$$\begin{aligned} \omega(\ell_a, n, \mathcal{K}; \lambda) &\leq \sup_{f_a, x_a \geq 0} \frac{(f_a - \lambda\, x_a)x_a + \frac{1}{2}(f_a)^2 - \frac{1}{2}\sum_{i=1}^{n}(f_a^i)^2 + \sum_{i=1}^{n}\theta_a^i}{(f_a)^2} \\ &\leq \tfrac{n-1}{2n} + \sup_{f_a, x_a \geq 0} \frac{(f_a - \lambda\, x_a)x_a + \sum_{i=1}^{n}\theta_a^i}{(f_a)^2}, \end{aligned} \tag{5.30}$$

where (5.30) follows from Cauchy-Schwarz inequality. Then, the proof proceeds along the lines of the proof of Lemma 5.36 except that we replace the factor $\frac{3}{2}$ by 1.

$$\omega(\ell_a, n, \mathcal{K}; \lambda) \leq \tfrac{n-1}{2n} + \max_{\mu \in [\frac{1}{\mathcal{K}}, 1]} (\tfrac{1}{4\lambda}) + (\tfrac{1}{4\lambda} - 1)\, \mu^2 + \tfrac{1}{2}\, \mu - \tfrac{(1-\mu)^2}{(\mathcal{K}-1)}. \tag{5.31}$$

Setting again $\lambda := 1$ yields

$$\omega(\ell_a, n, \mathcal{K}; 1) \leq \tfrac{n-1}{2n} + \max_{\mu \in [\frac{1}{\mathcal{K}}, 1]} \tfrac{1}{4} - \tfrac{3}{4}\mu^2 + \tfrac{1}{2\lambda}\mu - \tfrac{(1-\mu)^2}{(\mathcal{K}-1)}.$$

It is easy to see that $\mu = \frac{\mathcal{K}+3}{3\mathcal{K}+1}$ is optimal. Evaluating $\frac{1}{1-\omega(\mathcal{L}_1, n, \mathcal{K}, 1)}$ yields the desired bound. □

This bound is asymptotically 6-competitive. It provides, however, an explicit dependency on the number of games and players involved. For $n = 1$, we obtain a bound of $\frac{3\mathcal{K}+1}{2\mathcal{K}+2}$ for atomic network games with affine linear latency functions; this bound has previously been established by Cominetti, Correa and Stier-Moses [23]. For $\mathcal{K} \to \infty$ we trivially establish a bound that only depends on the number of games. If the latency functions of the OnlineNG are affine and if we allow for infinitely many atomic players, the online algorithm ASeqNash is $\frac{6n}{n+3}$-competitive. To see this, we calculate the limiting value in Corollary 5.39: $\lim_{\mathcal{K}\to\infty} \frac{2(3\mathcal{K}+1)n}{n\mathcal{K}+3n+3\mathcal{K}+1} = \frac{6n}{n+3}$. If we only have a single atomic player in each game, we can set $\mathcal{K} := n$ and evaluate $\frac{2(3\mathcal{K}+1)n}{n\mathcal{K}+3n+3\mathcal{K}+1}$.

Corollary 5.40. *If the latency functions of the* OnlineNG *are in $\mathcal{L}_1$ and we have one atomic player per game, the online algorithm* ASeqNash *is $\frac{6n^2+2n}{n^2+6n+1}$-competitive, where n is the total number of games.*

Now, we derive improved upper bounds that depend solely on n. We prove these bounds by designing and appropriate function $\lambda(n)$ with values in $\Lambda(\mathcal{L}_1, n, \mathcal{K})$.

Corollary 5.41. *If the latency functions of the* OnlineNG *are in $\mathcal{L}_1$, the competitive ratio of the online algorithm* ASeqNash *is at most*

$$\max\left\{T(n), \frac{3}{2}\right\},$$

where $T(n)$ is defined as

$$T(n) := \frac{\left(2n + \sqrt{2}\sqrt{n(3n+1)}\right)n\left(1 + 3n + \sqrt{2}\sqrt{n(3n+1)}\right)\sqrt{2}}{4\sqrt{n(3n+1)}\,(n+1)^2}.$$

Proof. For $\mathcal{K} \to \infty$ we have $\lim_{\mathcal{K}\to\infty} \frac{(1-\mu)^2}{(\mathcal{K}-1)} = 0$. Hence, (5.31) reduces to

$$\omega(\ell_a, n, \infty; \lambda) \leq \tfrac{n-1}{2n} + \max_{\mu \in [\frac{1}{\mathcal{K}}, 1]} \tfrac{1}{4\lambda} - (\tfrac{1}{4\lambda} - 1)\,\mu^2 + \tfrac{1}{2\lambda}\mu.$$

The maximization problem can be solved, leading to

$$\max_{\mu \in [\frac{1}{\mathcal{K}}, 1]} \tfrac{1}{4\lambda} - (\tfrac{1}{4\lambda} - 1)\,\mu^2 + \tfrac{1}{2\lambda}\mu \leq \frac{1}{4\lambda - 1}.$$

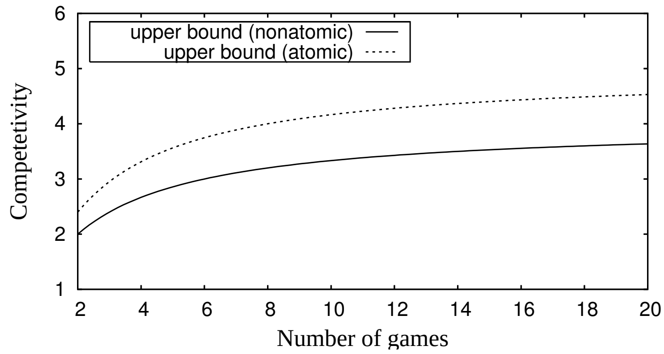

Figure 5.8: Upper bound $\frac{4n}{(n+2)}$ versus upper bound $T(n)$ on the competitive ratio of NSEQNASH and ASEQNASH for nonatomic and atomic players, respectively.

Defining

$$\lambda^* := \max\left\{\frac{1+3n+\sqrt{2n+6n^2}}{4(n+1)}, 1\right\},$$

It is easy to see that $\lambda^* \in \Lambda(\mathcal{L}_1, n, \mathcal{K})$. Applying Theorem 5.34 yields

$$C(f) \leq \frac{2\lambda^* n(-1+4\lambda^*)}{4n\lambda^* - 3n + 4\lambda^* - 1} C(x) = \max\left\{T(n), \frac{3}{2}\right\} C(x).$$

Taking x as the optimal offline solution proves the claim. □

Remark 5.42. The choice of λ^* solves the following minimization problem

$$\min_{\lambda \in \Lambda(\mathcal{L}_1, n, \mathcal{K})} \frac{2\lambda n(-1+4\lambda)}{4n\lambda - 3n + 4\lambda - 1}.$$

Note that without the restriction $\lambda \geq 1$, we have the relation $\lambda^* = \frac{1}{2} + \frac{1}{4}\sqrt{2}$ for $n = 1$ as in Corollary 4.22. For the case $n \to \infty$ we have $\lim_{n\to\infty} T(n) = \sqrt{2}\sqrt{3} + \frac{5}{2}$.

5.4.2 Lower Bounds

In this section, we provide lower bounds on the competitive ratio for any deterministic online algorithm and ASEQNASH. Note that all lower bounds of NSEQNASH for ONLINENG carry over to the atomic player case, if we allow for infinitely many players in each game i.

We use the network in Fig. 5.4 to derive a lower bound when we have a single atomic player in each game i.

Proposition 5.43. *In case of latency functions in $\mathcal{L}_1$, the online algorithm* ASEQNASH *for the* ONLINENG, *where in each game there is a single atomic player has a competitive ratio greater than or equal to $\frac{2n-1}{n}$, where n is the number of games.*

Table 5.3: Competitive Ratio for the online algorithm ASeqNash for affine linear latency functions $a_1\,x + a_0$, $a_1, a_0 \geq 0$. The first row shows known results for atomic network games that are due to Cominetti, Correa, and Stier-Moses [23]. UB and LB denote Upper and Lower Bound, respectively.

	arbitrary # of Players		1 player per game	
# games	UB	LB	UB	LB
1	$\frac{3}{2}$	1.343	1	1
2	$2\frac{2}{5}$	2	1.64	$\frac{3}{2}$
3	3	$2\frac{1}{3}$	2.14	$1\,\frac{2}{3}$
.	.	.	.	.
.	.	.	.	.
n	$\min\{T(n), 4.92\}$	$\frac{3n}{n-2}$	$\min\{\frac{6\,n^2+2\,n}{n^2+6\,n+1}, 4.92\}$	$\frac{2\,n-1}{n}$
∞	4.92	3	4.92	2

Proof. The proof proceeds along the lines of Theorem 5.17 except that we replace the constant costs $\ell_{(s_i,t_i)}(z) = 2\,i$, $i = 1, \ldots, k$. This forces the first k atomic players to route their demand along the middle arc (s,t). The reminder of the proof consists of technical calculations along the lines of the proof of Theorem 5.25. $\square$

Table 5.4.2 summarizes the main results for (affine) linear latency functions in this section.

5.4.3 General Latency Functions

In this section, we investigate the case, where we allow for general convex latency functions. Note that the only difference in the definition of $\omega(\ell_a; n, \mathcal{K}; \lambda)$ compared to the value $\omega(\ell_a; K, \lambda)$ introduced in (4.22) is the value $\vartheta_a^n(\ell_a, \boldsymbol{f}_a)$. By separating this value from the rest, we can rely on all characterizations of $\omega(\mathcal{L}; K, \lambda)$ obtained in Section 4.4.5.

Proposition 5.44. *If $\lambda \geq 0$ and $\ell_a(\cdot)$ is a nondecreasing latency function, then, the following inequality is valid:*

$$\omega(\ell_a; n, \mathcal{K}, \lambda) \leq \sup_{f_a, x_a \geq 0} \frac{\big(\ell_a(f_a) - \lambda\,\ell_a(x_a)\big)\,x_a + \ell_a'(f_a)\,\frac{x_a^2}{4}}{\ell_a(f_a)\,f_a} + \sup_{f_a \geq 0} \frac{\vartheta_a^n(\ell_a, \boldsymbol{f}_a)}{\ell_a(f_a)\,f_a}. \tag{5.32}$$

Proof. Using the triangle inequality, we can separate $\vartheta_a^n(\ell_a, \boldsymbol{f}_a)$ from the rest since the supremum over the sum of two functions is less than or equal to the sum of the suprema. The reminder of the proof follows Lemma 4.24. □

We define $\omega(\ell_a; \infty, \infty, \lambda)$ to be the limit of $\omega(\ell_a; n, \mathcal{K}, \lambda)$ for n and K tending to infinity under the condition that x_a and f_a are kept constant (and hence $\omega(\ell_a; n, \mathcal{K}, \lambda)$ stays finite). Then it follows that $\omega(\ell_a; n, \mathcal{K}, \lambda) \leq \omega(\ell_a; \infty, \infty, \lambda)$. We focus in the following on the general case $n, K \in \mathbb{N} \cup \{\infty\}$.

Corollary 5.45. *For latency functions in $\mathcal{L}_d$, $d \geq 1$, the competitive ratio of* ASeqNash *is at most*

$$\inf_{\lambda \in \Lambda(\mathcal{L}_d, n, \mathcal{K}) \cap \mathbb{R}^{\geq 1}} \left[\lambda \left(1 - \max_{0 \leq u \leq 1} u \left(1 - \lambda\, u^d + d\, \frac{u}{4}\right) + \frac{d}{d+1} \right)^{-1} \right]. \tag{5.33}$$

Proof. Recall from Lemma 5.21 that

$$\sup_{f_a \geq 0} \frac{\vartheta_a^n(\ell_a, \boldsymbol{f}_a)}{\ell_a(f_a)\, f_a} \leq \frac{d}{d+1}.$$

Hence, we only have to bound the first term in (5.33). Since all assumptions of Proposition 4.26 are satisfied with $s(f) = f^d$. Therefore, $s'(1) = d$ and

$$\omega(\ell_a; \infty, \infty, \lambda) \leq \max_{0 \leq u \leq 1} u \left(1 - \lambda\, u^d + d\, \frac{u}{4}\right) + \frac{d}{d+1}. \tag{5.34}$$

Applying Theorem 5.34 yields the claim. □

Using Corollary 5.45, we can determine bounds on the competitive ratio for ASeqNash for general polynomials. We use the same technique as in Theorem 4.28 to prove such bounds.

Theorem 5.46. *For latency functions in $\mathcal{L}_d$, $d \geq 1$, the competitive ratio of the online algorithm* ASeqNash *is at most*

$$\left(1 + \frac{5}{4}\, d + \frac{1}{4}\, d^2\right)^{d+1}.$$

Proof. Let the flow $\boldsymbol{f}$ be produced by the online algorithm ASeqNash and let $\boldsymbol{x}$ be an arbitrary feasible flow for the OnlineNG.

From Equation (5.34) in Proposition 5.45 we have the relation

$$\omega(\ell_a; \infty, \infty, \lambda) \leq \max_{0 \leq u \leq 1} u \left(1 - \lambda\, u^d + d\, \frac{u}{4}\right) + \frac{d}{d+1}.$$

Now, we bound the first term:

$$\max_{0 \leq u \leq 1} u \left(1 - \lambda\, u^d + d\, \frac{u}{4}\right) \leq \left(\frac{4+d}{4\,\lambda\,(d+1)} \right)^{\frac{1}{d}} \left(\frac{4\,d + d^2}{4\,(d+1)} \right).$$

We construct a function $\lambda(d)$ such that

$$\left(\frac{4+d}{4\,\lambda(d)\,(d+1)}\right)^{\frac{1}{d}}\left(\frac{4\,d+d^2}{4\,(d+1)}\right)=\frac{d}{(d+1)^2}$$

holds for all $d \geq 1$. Solving the above equation with respect to $\lambda(d)$ yields

$$\lambda^*(d)=\frac{(4+d)^{d+1}\,(d+1)^{d-1}}{4^{d+1}}.$$

Hence, by construction, we have

$$\omega(\ell_a;\infty,\infty,\lambda^*(d)) \leq \frac{d}{(d+1)^2}+\frac{d}{d+1}=\frac{d^2+2\,d}{(d+1)^2}.$$

Applying Theorem 5.34 with $\lambda := \lambda^*(d)$ and $\omega(\ell_a;\infty,\infty,\lambda^*(d)) \leq \frac{d^2+2\,d}{(d+1)^2}$ leads to

$$\begin{aligned} C(\boldsymbol{f}) &\leq \frac{\lambda^*(d)}{1-\frac{d^2+2\,d}{(d+1)^2}}\,C(\boldsymbol{x})=(d+1)^2\,\lambda^*(d)\,C(\boldsymbol{x}) \\ &= \big(1+\frac{d}{4}\big)^{d+1}\,(d+1)^{d+1}\,C(\boldsymbol{x})=\big(1+\frac{5}{4}\,d+\frac{1}{4}\,d^2\big)^{d+1}\,C(\boldsymbol{x}). \end{aligned}$$

Taking $\boldsymbol{x}$ as the optimal offline solution proves the claim. □

This result shows that the derived upper bounds are significantly larger than the bounds for the nonatomic counterpart. It is not clear, however, how to raise the lower bounds for the atomic case. This issue remains open.

Chapter 6

Conclusion and Open Issues

One of the main goals of this thesis was to understand the consequences of selfish behavior and limited knowledge about future information on the performance of routing strategies. We identified three practical applications for the considered models arising in road traffic networks and in the Internet.

First, we studied online routing strategies within the framework ONLINEMCRP that modeled the interactions of service providers in an inter-domain resource market. In such a market, network capacity is traded in order to deploy Internet traffic with Quality of Service requirements. We showed that a greedy online algorithm, which corresponds to a natural cost minimization strategy of a service provider, leads to a routing pattern that is not too inefficient. In particular, we showed that for polynomial price functions in $\mathcal{C}_d$, the competitive ratio of this greedy online algorithm can be bounded by a constant factor (depending on d) for arbitrary networks and commodity sequences. Even though the provable bounds are quite large, these bounds show that the proposed inter-domain market may not lead to arbitrary inefficient resource allocations. In practice, however, there are many more additional requirements to consider. For instance, routings have to respect capacities, which we only incorporated implicitly using steep load dependent price functions. With capacities, however, one can easily construct examples in which any online algorithm does not even produce a feasible solution. Further requirements in practice include path length restrictions and survivability issues. Another important point is that in practice, routings are only valid until a given time, after which they disappear. This has effects on the cost for future routings. It is also an open issue whether the competitiveness bound in Theorem 3.9 and Theorem 3.24 are tight, and whether there exists a competitive online algorithm for the unsplittable variant of the ONLINEMCRP. Finally, we simplified the competition within the market by assuming fixed continuous and non-decreasing price functions defining the price for a unit resource. In practice, resource providers determine prices depending on the current market situation and their position with respect to the network topology. If the provider domain's link is a bottleneck, the demand would become somewhat inelastic leading to a monopolistic situation. For a fully connected network (i.e. perfect competition in the network), the demand is at a minimum when the offered price is above the current market price and at maximum when below. The infrastructure of the Internet today is more related to an oligopolistic market where the network is not fully connected (i.e. domains are at most connected to 3 to 5 neighboring domains). We are only aware of few works on this complex topic. Acemoglu and Ozdaglar [2] study the competition of service providers for very simple network topologies such as parallel arcs or serial arcs. In this regard, the outcome of competition between service providers for general network topologies, where demand is elastic remains tantalizing open. We note that the above issues are closely related to the theory of *mechanism design*. Mechanism design is a subfield of economics and computer science that aims at designing the rules of games involving multiple competing players to achieve a pre-specified outcome. For an introduction into this topic, we refer to the survey of Mas-Colell, Whinston, and Green [66]. In the context of network design problems we refer to Gupta, Srinivasan, and Tardos [49], Anshelevich et al. [6, 7], Archer et al. [8], and Fleischer et al. [36].

The second application that motivated the second main contribution of this thesis concerns the road traffic network in which drivers select routes based on selfish interests. A long standing open question asked to which extent the performance of a Nash equilibrium is degraded compared to the system optimum. This question has been settled for the nonatomic traffic model by Roughgarden and Tardos [84] and Roughgarden in [81]. For network games with atomic players we contributed to answering the same question by improving previous known bounds on the price of anarchy for polynomial latency functions in $\mathcal{L}_d$, except for the case $\mathcal{L}_1$. These results are of particular interest as a recent trend towards using route guidance devices can be observed. Such intelligent transport systems control an atomic part of the entire traffic demand. Therefore, the framework of atomic network games applies in this case. Even though Cominetti, Correa, and Stier-Moses in [23] present instances, where the performance of a Nash equilibrium deteriorates compared to the Nash equilibrium of the corresponding nonatomic network game, our results show that the efficiency loss is still bounded by a constant factor depending on d for polynomial latency functions in $\mathcal{L}_d$. We note that still all known lower bounds do not match the upper ones. We see this mismatch as an important open issue to be resolved. Besides network games in which players seek to route given demands resource allocation games in networks with elastic demands have recently gained much attention. For a simple resource allocation mechanism, Kelly, Maulloo, and Tan [58] and Gibbens and Kelly [45] showed that for price taking players the outcome of a Nash equilibrium is optimal. Johari and Tsitsiklis [56, 57] showed that for price anticipating players the inefficiency of the Nash equilibrium is bounded by 33% compared to the system optimal resource allocation.

Finally, we studied the source routing model for the Internet. We identified a major drawback of the underlying equilibrium concept: In order to converge to an equilibrium, traffic sources have to be aware of traffic changes within the network to react accordingly. This implies that sources have to maintain the state of all available routes during the entire connection duration. Therefore, in addition to the regular payload the total traffic volume is blown up by continuously signaling this needed information. We investigated a different model in which sources select routes only during a predefined time frame. We simplified the analysis by assuming that groups of sources have different release times and every group of sources converges to a Nash equilibrium before the next release time. In this regard, we considered a sequence of games (groups) in which sources once they are at Nash equilibrium fix their routings. By combining methods from online optimization with methods from algorithmic game theory we showed that the competitive ratio of the online algorithm NSeqNash and ASeqNash, which produce a flow that is at Nash equilibrium for every game can be bounded by a constant factor (depending on d) for polynomial latency functions in $\mathcal{L}_d$. By definition of the algorithms, these results hold for nonatomic as well as atomic player types. Even though these results indicate that the proposed working mechanism is quite efficient, our simplifying assumptions are still to far away from reality. In practice, sources and groups of sources start at arbitrary release times. Moreover, if we assume that every source changes the routing only during a given time frame it is not granted that the entire system is at equilibrium at any point in time, see Even-Dar and Mansour [31] for work on the convergence speed. Nevertheless, we believe that we can achieve an accurate approximation of the dynamics of a real system by considering a sequence of Nash equilibria over time. It is open, however, if it is possible to prove exact approximation guarantees in this case. As for the OnlineMCRP, open issues also include arc capacities and different expiring times for the demands.

Bibliography

[1] K. Abbink and J. Brandts. Price competition under cost uncertainty: A laboratory analysis. *UFAE and IAE Working Papers 550.02*, 2002. (Cited on page 17.)

[2] D. Acemoglu and A. Ozdaglar. Competition in parallel-serial networks. *IEEE Journal on Selected Areas in Communications, Special Issue on Non-Cooperative Behavior in Networking*, 2006. forthcoming. (Cited on page 116.)

[3] S. Aland, D. Dumrauf, M. Gairing, B. Monien, and F. Schoppmann. Exact price of anarchy for polynomial congestion games. In *STACS*, pages 218–229, 2006. (Cited on page 58.)

[4] S. Albers. Online Algorithms: A Survey. *Mathematical Programming*, 97 (1–2):3–26, July 2003. (Cited on page 16.)

[5] E. Altman, T. Basar, T. Jimenez, and N. Shimkin. Competitive routing in networks with polynomial costs. *IEEE Transactions on Automatic Control*, 47(1):92–96, 2002. (Cited on pages 9, 19, and 81.)

[6] E. Anshelevich, A. Dasgupta, E. Tardos, and T. Wexler. Near-optimal network design with selfish agents. In *Proc. 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 511–520, 2003. (Cited on page 116.)

[7] E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, T. Wexler, and T. Roughgarden. The price of stability for network design with fair cost allocation. In *Proc. 45th Annual IEEE Symposium on Foundations of Compututer Science (FOCS)*, 2004. (Cited on page 116.)

[8] A. Archer, J. Feigenbaum, A. Krishnamurthy, R. Sami, and S. Shenker. Approximation and collusion in multicast cost sharing. *GEB*, 47:36–71, 2004. (Cited on page 116.)

[9] B. Awerbuch, Y. Azar, and S. Plotkin. Throughput-competitive on-line routing. In *Proceedings of the 34th Annual IEEE Symposium on Foundations of Computer Science (FOCS), Palo Alto*, pages 32–40, 1993. (Cited on page 19.)

[10] B. Awerbuch, Y. Azar, and A. Epstein. The price of routing unsplittable flow. In *Proceedings of the 37th annual ACM symposium on Theory of Computing (STOC)*, pages 57–66, New York, NY, USA, 2005. ACM Press. (Cited on page 58.)

[11] M. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the Economics and Transportation*. Yale University Press, 1956. (Cited on pages 8 and 57.)

[12] S. Ben-David, A. Borodin, R. M. Karp, G. Tardos, and A. Wigderson. On the Power of Randomization in On-Line Algorithms. *Algorithmica*, 11(1): 2–14, 1994. (Cited on pages 13 and 16.)

[13] P. Bergendorff, D. Hearn, and M. Ramana. Congestion Toll Pricing of Traffic Networks. In D. H. P. M. Pardalos and W. Hager, editors, *Network Optimization*, volume 450 of *LNCS*, pages 51–71. Springer-Verlag, Berlin, 1997. (Cited on page 60.)

[14] A. Blum, E. Even-Dar, and K. Ligett. Routing without regret: on convergence to nash equilibria of regret-minimizing algorithms in routing games. In *PODC*, pages 45–52, 2006. (Cited on page 83.)

[15] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998. (Cited on pages 14, 16, and 25.)

[16] D. Braess. Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung*, 12:258–268, 1969. (Cited on page 57.)

[17] D. Branston. Link capacity functions. *Transportation Research*, 10:223–236, 1976. (Cited on pages 46, 55, and 58.)

[18] Bureau of Public Roads. Traffic assignment manual. *U.S. Department of Commerce, Urban Planning Division, Washington, DC*, 1964. (Cited on page 57.)

[19] S. Catoni and S. Pallotino. Traffic equilibrium paradoxes. *Transportation Science*, 25:240–244, 1991. (Cited on pages 8, 55, and 58.)

[20] G. Christodoulou and E. Koutsoupias. The price of anarchy of finite congestion games. In *Proceedings of the 37th annual ACM symposium on Theory of Computing (STOC)*, pages 67–73, New York, NY, USA, 2005. ACM Press. (Cited on page 58.)

[21] R. Cole, Y. Dodis, and T. Roughgarden. Pricing network edges for heterogeneous selfish users. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 521–530, 2003. (Cited on page 58.)

[22] R. Cole, Y. Dodis, and T. Roughgarden. How much can taxes help selfish routing? In *Proc. 4th ACM Conference on Electronic Commerce*, pages 98–107, 2003. (Cited on page 58.)

[23] R. Cominetti, J. Correa, and N. Stier-Moses. Network games with atomic players. In *Proceedings of the 33rd International Colloquium of Automata, Languages and Programming (ICALP'06)*, 2006. (Cited on pages 8, 58, 59, 70, 71, 72, 73, 74, 76, 77, 79, 106, 109, 111, and 116.)

[24] J. R. Correa, A. S. Schulz, and N. E. Stier Moses. Selfish routing in capacitated networks. *Math. Oper. Res.*, 29:961–976, 2004. (Cited on pages 8, 19, 34, 57, 59, 64, 65, 66, and 102.)

[25] J. R. Correa, A. S. Schulz, and N. E. S. Moses. On the inefficiency of equilibria in congestion games. In M. Jünger and V. Kaibel, editors, *Integer programming and combinatorial optimization. 11th international IPCO conference, Berlin*, volume 3509 of *LNCS*, pages 167–181. Springer, Berlin, 2005. (Cited on pages 8, 57, 58, 59, and 70.)

[26] A. Czumaj and B. Vöcking. Tight bounds for worst-case equilibria. In *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms (SODA)*, pages 413–420, Philadelphia, PA, USA, 2002. Society for Industrial and Applied Mathematics. (Cited on page 57.)

[27] S. Dafermos and F. Sparrow. The traffic assignment problem for a general network. *J. Res. Natl. Bur. Stand., Sect. B*, 73:91–118, 1969. (Cited on pages 23, 62, 63, 66, 68, 69, 70, and 87.)

[28] P. Dubey. Inefficiency of Nash Equilibria. *Math. Oper. Res.*, 11:1–8, 1986. (Cited on page 57.)

[29] D. Dumrauf and M. Gairing. Price of anarchy for polynomial wardrop games. In *WINE*, pages 319–330, 2006. (Cited on page 57.)

[30] R. Engelberg and J. S. Naor. Equilibria in online games. In *Proceedings of the 18th annual ACM-SIAM Symposium on Discrete Algorithms (SODA'07)*, pages 149–158, 2007. (Cited on page 83.)

[31] E. Even-Dar and Y. Mansour. Fast convergence of selfish rerouting. In *Proceedings of the 16th Annual ACM–SIAM Symposium on Discrete Algorithms (SODA)*, pages 772–781, 2005. (Cited on pages 81, 83, and 117.)

[32] A. Fiat and G. J. Woeginger, editors. *Online Algorithms: The State of the Art*, volume 1442 of *Lecture Notes in Computer Science*. Springer, 1998. (Cited on pages 19 and 25.)

[33] S. Fischer and B. Vöcking. Adaptive routing with stale information. In M. K. Aguilera and J. Aspnes, editors, *Proc. 24th Ann. ACM SIGACT-SIGOPS Symp. on Principles of Distributed Computing (PODC)*, pages 276–283, Las Vegas, NV, USA, July 2005. ACM. doi: 10.1145/1073814.1073868. (Cited on page 83.)

[34] S. Fischer, H. Räcke, and B. Vöcking. Fast convergence to Wardrop equilibria by adaptive sampling methods. In *Proc. 38th Ann. ACM. Symp. on Theory of Comput. (STOC)*, pages 653–662, Seattle, WA, USA, May 2006. ACM. doi: 10.1145/1132516.1132608. (Cited on page 83.)

[35] L. Fleischer, K. Jain, and M. Mahdian. Tolls for heterogeneous selfish users in multicommodity networks and generalized congestion games. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 277–285, Washington, DC, USA, 2004. IEEE Computer Society. (Cited on page 58.)

[36] L. Fleischer, J. Könemann, S. Leonardi, and G. Schäfer. Simple cost sharing schemes for multicommodity rent-or-buy and stochastic steiner tree. In *Proceedings of the 38th annual ACM symposium on Theory of Computing (STOC)*, pages 663–670, New York, NY, USA, 2006. (Cited on page 116.)

[37] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. In *Proceedings of IEEE INFOCOM*, pages 519–528, Tel-Aviv, Israel, March 2000. (Cited on page 46.)

[38] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS weights in a changing world. *IEEE JSAC*, 20(4):756–767, 2002. (Cited on page 18.)

[39] B. Fortz and M. Thorup. Increasing internet capacity using local search. *Computational Optimization and Applications*, 29(1):13–48, 2004. (Cited on page 18.)

[40] D. Fotakis, S. C. Kontogiannis, and P. G. Spirakis. Atomic congestion games among coalitions. In *Proceedings of the 33rd International Colloquium of Automata, Languages and Programming (ICALP'06)*, pages 572–583, 2006. (Cited on page 58.)

[41] Frame Relay Forum. FRF.13 service level definitions implementation agreement, August 1998. (Cited on page 45.)

[42] E. Friedman. Genericity and congestion control in selfish routing. In *43rd IEEE Conference on Decision and Control*, pages 4667 – 4672, 2004. (Cited on pages 9 and 81.)

[43] M. R. Garey and D. S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979. (Cited on page 41.)

[44] H. Gharavi and M. H. Partovi. Multilevel video coding and distribution architectures for emerging broadband digital networks. *IEEE Trans. on Circuits and Systems for Video Technology*, 6(5):459–469, Oktober 1996. (Cited on page 17.)

[45] R. Gibbens and F. Kelly. Resource pricing and the evolution of congestion control. *Automatica*, 35:1969–1985, 1999. (Cited on page 116.)

[46] D. Goderis, D. Griffin, C. Jacquenet, and G. Pavlou. Attributes of a service level specification (SLS) template. *Internet-Draft draft-tequila-sls-03.txt*, October 2003. (Cited on page 45.)

[47] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1994. (Cited on page 38.)

[48] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics*. Springer-Verlag, Heidelberg, 2nd edition, 1993. (Cited on pages 23, 24, 47, and 62.)

[49] A. Gupta, A. Srinivasan, and E. Tardos. Cost-sharing mechanisms for network design. In *Proc. 7th Int. Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX)*, pages 139–150, 2004. (Cited on page 116.)

[50] A. Hall, K. Langkau, and M. Skutella. An fptas for quickest multicommodity flows with inflow-dependent transit times. *Alorithmica*, to appear, 2007. (Cited on page 58.)

[51] T. Harks and L. A. Végh. Selfish routing with online demands. In *Proceedings of the Fourth Workshop on Combinatorial and Algorithmic Aspects of Networkin (CAAN), forthcoming*, 2007. (Cited on page 84.)

[52] T. Harks, S. Heinz, and M. E. Pfetsch. Online multicommodity routing problem. In *Proceedings of Fourth Workshop on Approximation and Online Algorithms (WAOA 2006)*, 2006. (Cited on pages 20 and 82.)

[53] A. Hayrapetyan, Tardos, and T. Wexler. The effect of collusion in congestion games. In *Proceedings of the 38th annual ACM symposium on Theory of Computing (STOC)*, pages 89–98, New York, NY, USA, 2006. ACM Press. (Cited on page 58.)

[54] D. Hearn and M. Ramana. Solving congestion toll pricing models. In P. Marcotte and S. Nguyen, editors, *Equilibrium and Advanced Transportation Modeling*, pages 109–124. Kluwer Academic Publishers, 1998. (Cited on page 58.)

[55] O. Jahn, R. H. Möhring, A. S. Schulz, and N. E. Stier-Moses. System-optimal routing of traffic flows with user constraints in networks with congestion. *Oper. Res.*, 53:600–616, 2005. (Cited on page 58.)

[56] R. Johari and J. N. Tsitsiklis. Efficiency loss in a network resource allocation game. *Math. Oper. Res.*, 29(3):407–435, 2004. (Cited on page 116.)

[57] R. Johari and J. N. Tsitsiklis. A scalable network resource allocation mechanism with bounded efficiency loss. *IEEE JSAC Special issue: Price-Based Access Control and Economics for Communication Networks.*, 24(5):992–999, 2006. (Cited on page 116.)

[58] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan. Rate Control in Communication Networks: Shadow Prices, Proportional Fairness, and Stability. *Journal of the Operational Research Society*, 49:237–52, 1998. (Cited on page 116.)

[59] J. M. Kleinberg. Single-source unsplittable flow. In *Proceedings of 37th IEEE Conference on Foundations of Computer Science (FOCS)*, Burlington, VT, USA, October 1996. (Cited on page 47.)

[60] T. Koch. *Rapid Mathematical Programming*. PhD thesis, Technische Universität Berlin, 2004. ZIB-Report 04-58. (Cited on page 49.)

[61] E. Köhler and M. Skutella. Flows over time with load-dependent transit times. *SIAM J. on Optimization*, 15(4):1185–1202, 2005. (Cited on page 58.)

[62] E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In C. Meines and S. Tison, editors, *Proc. of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 1563 of *LNCS*, pages 404–413. Springer, Trier, 1999. (Cited on pages 5, 57, and 63.)

[63] R. La, J. Walrand, and V. Anantharam. Issues in TCP Vegas. Electronics Research Laboratory, University of California, Berkeley, UCB/ERL Memorandum, No. M99/3, January 1999. (Cited on pages 10 and 81.)

[64] T. Larsson and M. Patriksson. Side constrained traffic equilibrium models—analysis, computation and applications. *Transportation Research*, 33B:233–264, 1999. (Cited on page 57.)

[65] P. Marcotte, S. Nguyen, and A. Schoeb. A strategic flow model of traffic assignment in static capacitated networks. *Oper. Res.*, 52(2):191–212, 2004. (Cited on page 57.)

[66] Mas-Colell, Andreu, Whinston, M. D., and J. R. Green. *Microeconomic Theory*. Oxford University Press, 1995. (Cited on page 116.)

[67] I. Milchtaich. Generic uniqueness of equilibrium in large crowding games. *Math. Oper. Res.*, 25(3):349–364, 2000. (Cited on page 55.)

[68] I. Milchtaich. Congestion games with player-specific payoff functions. *Games and Economic Behavior*, 13(1):111–124, 1996. (Cited on page 58.)

[69] I. Milchtaich. Topological conditions for uniqueness of equilibrium in networks. *Math. Oper. Res.*, 30(1):225–244, 2005. (Cited on page 58.)

[70] R. Motwani and P. Raghaven. *Randomized Algorithms*. Cambridge University Press, Cambridge UK, 1995. (Cited on page 16.)

[71] J. Nash. Non-cooperative games. *The Annals of Mathematics*, 54(2):286–295, September 1951. (Cited on pages 56 and 61.)

[72] A. Orda, R. Rom, and N. Shimkin. Competitive routing in multiuser communication networks. *IEEE/ACM Transactions on Networking*, 1(5):510–521, 1993. (Cited on pages 55 and 58.)

[73] M. Patriksson. *The Traffic Assignment Problem—Models and Methods*. Topics in Transportation, VSP, 1994. (Cited on pages 46, 55, and 57.)

[74] G. Perakis. The price of anarchy when costs are non-separable and asymmetric. In D. Bienstock and G. Nemhauser, editors, *Proc. 10th International Conference on Integer Programming and Combinatorial Optimization, (IPCO), New York*, volume 3064 of *Lecture Notes in Computer Science*, pages 46–58. Springer, Berlin Heidelberg, 2004. (Cited on page 57.)

[75] A. C. Pigou. *The Economics of Welfare*. Macmillan, 1920. (Cited on pages 8, 56, and 103.)

[76] L. Qiu, R. Yang, Y. Zhang, and S. Shenker. On selfish routing in internet-like environments. *IEEE/ACM Transactions on Networking*, 14, 4:725–738, 2006. (Cited on pages 9 and 81.)

[77] O. Richman and N. Shimkin. Topological uniqueness of the nash equilibrium for atomic selfish routing. *Math. Oper. Res.*, forthcoming, 2007. (Cited on page 58.)

[78] J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica*, 33:520–534, 1965. (Cited on pages 8, 58, 70, 81, and 88.)

[79] R. W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973. (Cited on page 55.)

[80] T. Roughgarden. *Selfish Routing*. PhD thesis, Cornell University, 2002. (Cited on pages 9, 57, and 81.)

[81] T. Roughgarden. The price of anarchy is independent of the network topology. *Journal of Computer and System Science*, 67:341–364, 2002. (Cited on pages 8, 57, 59, 63, 64, 65, 66, 79, and 116.)

[82] T. Roughgarden. *Selfish Routing and the Price of Anarchy*. The MIT Press, 2005. (Cited on page 61.)

[83] T. Roughgarden. Selfish routing with atomic players. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 973–974, 2005. (Cited on pages 8, 58, 59, 70, 71, and 72.)

[84] T. Roughgarden and E. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002. (Cited on pages 8, 19, 34, 55, 57, 58, 61, 63, 66, 69, 83, 86, 97, and 116.)

[85] A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*, volume 24 of *Algorithms and Combinatorics*. Springer, Berlin Heidelberg, 2003. (Cited on page 42.)

[86] TeleManagement Forum. GB917 Eva.Ver.1.5 SLA management handbook, June 2001. (Cited on page 45.)

[87] J. G. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineers*, 1(Part II):325–378, 1952. (Cited on pages 8, 56, and 61.)

[88] A. Yahaya and T. Suda. iREX: Inter-domain resource exchange architecture. In *Proceedings of IEEE INFOCOM*, April 2006. (Cited on pages 6, 17, 18, 21, 45, and 48.)

[89] A. Yahaya and T. Suda. iREX: Inter-domain QoS automation using economics. In *Proceedings of IEEE Consumer Communications and Networking Conference*, Jan 2006. (Cited on pages 17, 18, 21, 45, and 48.)

[90] A. Yahaya, T. Harks, and T. Suda. iREX: Efficient inter-domain QoS automation using economics. In *Proceedings of IEEE Globecom*, November 2006. (Cited on pages 6, 17, and 20.)