SELFISH BEHAVIOR IN NETWORK-BASED GAMES

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Thomas B. Wexler

January 2006

SELFISH BEHAVIOR IN NETWORK-BASED GAMES

Thomas B. Wexler, Ph.D.

Cornell University 2006

It is a well known pricinple of economics that when members of society do not coordinate and instead act independently in a self-interested fashion, the resulting outcome typically does not achieve the maximum possible total social welfare. Thus, there is often a cost to competitive behavior. This phenomenon, often refered to as the “Tragedy of the Commons”, can explain the underlying cause of many diverse problems, including industrial over-pollution, nuclear proliferation, and over-fishing. In all these settings, decentralized and selfish behavior leads to suboptiml outcomes in which all parties suffer. In this thesis, we explore the extent to which a similar lack of centralized control in the design, formation, and operation of the Internet might likewise lead to an inefficient, unreliable, and needlessly expensive network.

To study such scenarios, we apply the tools of game theory. This branch of mathematics allows us to formally describe these competitve environments, which we call *games*. Self-interested agents are represented as *players*, who seek to maximize their own private welfare. The stable solutions reached by these competing players are called *Nash equilibria*. In this thesis, we consider a number of games which model various aspects of competition in the Internet, and seek to quantify the extent to which the lack of centralized control decreases social welfare. In particular, we study games that model the competitive creation of networks and the pricing of network services, and we examine how collusion can make matters even worse. Our primary goal is to compare the

quality of Nash equilibria to that of centrally designed solutions.

We first introduce a game that models network formation. We show that while worst case selfish behavior is very costly, in a broad class of games, the best stable solutions are just as good as those that are designed centrally. In other words, for a large class of network creation games, selfish behavior need not lessen social welfare. We then consider the computational complexity of finding such stable outcomes. We also explore how minor resitrictions on players' choices can lead to dramatically better outcomes. In particular, we show that by enforcing a certain degree of "fairness" in our network formation game, the cost of solutions reached through selfish play improves dramatically.

We next consider a game in which players set prices for network services, and compete with each other for customers. We provide conditions under which Nash equilibria are guaranteed to exist, and show that for a broader class of games, equilibria are always close to optimal. Lastly, we propose a framework to model the formation of coalitions, and study these coalitions in the context of a simple load-balancing game. We show that despite the apparent increase in the level of coordination, limited collusion can actually decrease the quality of solutions that are reachable through selfish play. However, we prove that for a large class of games, this increase is always small.

BIOGRAPHICAL SKETCH

Tom Wexler was born on August 6th, 1978. He recieved a B.A. in Mathematics and Computer Science from Amherst College in May 2000, an M.S. in Computer Science from Cornell University in May 2004, and expects to receive a Ph.D. from Cornell University in January 2006.

To my mom, with love.

ACKNOWLEDGEMENTS

First and foremost I would like to thank my advisor, Éva Tardos. Her guidance, support, and encouragement have been invaluable to me. I would like to thank David Shmoys, who, beyond teaching some of the best classes I've had here at Cornell, has also been most helpful with his advice on research and giving talks. And I'd like to thank Jon Kleinberg; working with him has been very enjoyable, and I have been inspiried by his uncanny ability to both simplify the most complex of problems and draw connections between distant areas.

I would also like to thank those who advised me at Amherst; David Armacost, John Rager, and Norton Starr. David is responsible for first getting me hooked on mathematics, with his introductory algebra course. John introduced me to computer science, and was the first to encourage me to go to graduate school. And Norton has always been supportive of my efforts, both inside the classroom and out. I would also like to thank Ruth Haas of Smith College, for advising me on my undergraduate thesis. And I must thank David Kelly of Hampshire College. Through the Hampshire College Summer Studies in Mathematics program, Kelly introduced me to the joy of teaching math, and for that I am forever grateful.

While there are many people at Cornell who have helped me great deal, I am particularly indebted to three; Becky Steward, Stephanie Meik, and Cindy Robinson. Without their help, friendship, and infinite patience, I would have simply not made it. Maybe a year or two, tops. But that's it. I hope you're willing to put up with me for one more year. I'll stop bothering you after that, I promise.

To my friends and family, your support has meant more than you'll ever know.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Overview

Traditional network design seeks to construct and operate networks so as to optimize some feature or features of the network, such as cost, average delay, congestion, or connectivity. Although there are many well studied and understood variants of this problem, nearly all work in the field operates under the assumption that a single authority is responsible for constructing an optimal (or near-optimal) solution. In other words, a single centralized designer is presumed to exist, with the ability to single-handedly construct a solution, and an interest only in optimizing the overall quality of the system. However, many real large-scale networks, most notably the Internet, do not fall within this framework. The construction and operation of such networks are not orchestrated by some benevolent and centralized authority. Instead, these networks evolve through the interaction, both competitive and collaborative, of self-interested agents. Each agent has some limited and local objectives that he is interested in optimizing. Collectively, the goals of the agents may coincide with the objectives considered in traditional network design, but individually, each agent is unconcerned with any notion of a global objective function.

For example, consider the generalized Steiner forest problem; given a network with edge costs and a list of terminal pairs, we seek to select the cheapest edge-set that connects all given pairs. This is a classical problem, and has been well studied as a simple model of cost-efficient network design. But consider what happens if instead of having a central designer, we have distinct agents, each responsible for connecting a single terminal pair as cheaply as possible. In this case, there may be solutions that are globally very cheap, but require the cooperation of many agents. If some of these agents have alternate routes available to them, they may abandon the remaining agents to save on cost. The remaining agents, however, may not have alternative options available, and now a smaller number of them must cover the same cost as before. Hence competitive play can settle on a solution that is more expensive than one designed by a central authority. The primary goal of this thesis is to understand the price of decentralizing this and related network-based problems.

Game theory provides a natural tool to study such a scenario. By modeling the behavior of self-interested agents as a game, we can explore the stable outcomes – the Nash equilibria – that may result from these competitive interactions. Our primary interest lies in studying the quality of these solutions, which in the above example would best be defined as the cost of the resulting network. Game theory is full of examples in which the self interested behavior of agents leads to a "tragedy of the commons" [31], in which all players fare far worse than they would have in a centrally dictated solution (The Prisoners' Dilemma, described in chapter 2, is perhaps the best known example of this). Hence it is reasonable to try to quantify the cost of selfish behavior. How much more will selfish agents spend than would a single centralized designer? More precisely, we are interested in measuring the *price of anarchy*, as introduced by Koutsoupias and Papadimitriou [45, 54], which compares the value of the worst-case Nash equilibria to that of the optimal solution. A low price of anarchy indicates that completely unregulated and selfish behavior does not cause a tragedy of the commons; the resulting stable outcome may not be optimal, but it will be nearly so.

Another important issue to explore in this area is the middle ground between centrally enforced solutions and completely unregulated anarchy. In most networking applications, it is not the case that agents are completely unrestricted; rather, they interact with an underlying protocol that essentially proposes a collective solution to all participants, who can each either accept it or defect from it. As a result, it is in the interest of the protocol designer to seek the *best* Nash equilibrium; this can naturally be viewed as the optimum, subject to the constraint that the solution be *stable*, with no agent having an incentive to unilaterally defect from it once it is offered. Hence, one can view the ratio of the solution quality at the best Nash equilibrium relative to the global optimum as a *price of stability*, since it captures the problem of optimization subject to this constraint. Some of our recent work with Anshelevich, Dasgupta, Kleinberg, Tardos, and Roughgarden [8, 7], as well as research by Correa, Schulz, and Stier Moses [17], has explored this definition (which we termed the "optimistic price of anarchy" in [8]); it stands in contrast to the larger line of work in algorithmic game theory on the price of anarchy, which is more suited to worst-case analysis of situations with essentially no protocol mediating interactions among the agents. Indeed, one can view the activity of a protocol designer seeking a good Nash equilibrium as being aligned with the general goals of mechanism design — producing a game that yields good outcomes when players act in their own self-interest.

This thesis will examine the behavior of selfish agents in networks. We will consider an abstracted view of the operation and development of networks, based primarily on the Internet. In particular, we consider the Internet as being designed, built, and maintained by numerous and distinct Autonomous Systems. These economically-minded interests are responsible for creating the main backbone links that connect distant servers and local area networks. Acting as service providers, these companies charge network users for access to the connections they have formed. The network users then choose between competing service providers, and decide how to route their own traffic in the given network. Like the autonomous systems, network users are also economically motivated, and are sensitive to both prices and the quality of service they recieve.

In this thesis, we consider games that seek to model network formation, the pricing of networks for selfish traffic, and collusion among players in network-based settings. The focus of our study is the price of anarchy and price of stability in these games, but we also consider other issues such as computability, convergence of natural play, and the effects of forming coalitions in games.

1.2 Models and Results

Here we provide an informal summary of the models and results presented in this thesis.

1.2.1 Competitive Network Formation

We first consider two related games that model the construction of a network by independent, competitive agents. In these games, there are k players, each of whom has a set of specified terminals in an underlying network. Each players selects a tree connecting his terminals. Players must pay for the edges in their tree, and seek to minimize their payments. When multiple players use an edge, they may share its cost. Thus, agents have an incentive to cooperate with each other in building a shared network. However, they will not cooperate if they can lower their expenses by joining their terminals along a less collaborative yet cheaper tree. To complete the description of the game, we need only define how players may share edge costs. The two games we consider correspond to two different sharing rules.

We formally introduce this model in chapter 3, and consider the variant in which the sharing of edge costs is completely unregulated; any division of cost among players using an edge is acceptable so long as the edge is fully paid for. More formally, every agent specifies payments towards the edges in their selected tree. They seek to pay as little as possible yet must ensure that all edges in their tree are fully paid for. We refer to this as *unrestricted cost sharing*. We show that even among such games with only two players, each with only two terminals, there are examples in which no pure Nash equilibrium exists. Furthermore, even when equilibria exist, the price of stability may be $O(k)$.

In light of this, we consider approximate equilibria. We say that a solution is an α*-approximate Nash equilibrium* if no player can improve their utility by more than a factor of α by deviating from their current strategy. We show that in any game, there is always a 3-approximate Nash equilibrium that purchases the optimal network. Further, we prove that this guarantee is within a factor of two from the best possible bound, and introduce efficient algorithms that produce solutions nearly matching our guarantee.

We also consider a class of instances where much stronger positive results apply. We call a game *single-source* if all players share a single fixed terminal and have one additional terminal. Such a game can be viewed as modeling agents attempting to build connections to a central server. We prove that in single-source games, not only do Nash equilibria always exist, but there are always equilibria that build the optimal network. While finding such a network is NP-complete, we show that given any good approximation to the optimal Steiner tree, we can efficiently construct a solution that is arbitrarily close to a Nash equilibrium of lesser or equal cost. A useful interpretation of this is that a central authority can subsidize a very small portion of the desired network, and the players will be content to purchase the remainder.

However, both the lack of equilibria in general games and the potential for a high price of stability are troubling. To address these problems, we consider limiting players' strategies. In chapter 4 we provide a dramatic demonstration of the power of mechanism design. We show that we can guarantee that pure Nash equilibria exist simply by requiring that players sharing an edge divide up its cost evenly, which we refer to as *fair cost sharing*. Furthermore, under the fair sharing rule, the price of stability in general games drops from k to $\log(k)$, a bound we prove to be tight. Thus, with only a minor and arguably fair restriction, the quality of outcomes in this game improve drastically.

This model also admits a number of natural extensions. In particular, we generalize this game to a set-based (as opposed to network-based) game. From a networking perspective, such a model encompasses games where players have stronger connectivity requirements. We extend our results to models in which edge costs are not constant, but instead are more realistic cost functions that increase with the number of users in a buy-at-bulk fashion. We also consider the addition of congestion based latencies to edge costs, convergence issues, and a weighted generalization of the game.

1.2.2 Pricing Networks for Selfish Users

In the two models dicussed so far, the explicit goal of the players constructing the network is to connect their terminals. In reality, the primary objective of the corporations that build large-scale networks is to generating revenue by acting as service providers and charging users for network access. These companies are only interested in the connections they create to the extent that they are profitable. In particular, this suggests that if we are to model how networks are built and designed, we must first understand the role played by the potential network users. Presumably such users are themselves rational, self-interested agents, who seek service that is both cheap and fast. How do network designers behave when both their clients and their competition act in a competitive fashion? Our next step is to develop a model that captures the interaction between such price-setting service providers and the selfish network users they serve.

We initiate the study of pricing networks for selfish users in chapter 5 with a simple game played on a network of parallel links. Players in this game are service providers who compete for network users. Each player controls a single link, and charges users for access to that link. Each link has a latency function that specifies how the latency experienced by traffic on a link increases as the volume of traffic on that link increases. Once prices on links are set, network users select a service provider based on price and latency. The total volume of user demand depends on quality service in the natural way; as latency and cost increase, demand for network access decreases. Players gather profit based on the prices they set and the resulting volume of traffic that chooses to use their links.

We are again interested in the level of social welfare attained when all parties act in a self-interested manner. However, unlike in the previous two chapters, total social welfare can not be fully captured by only considering the profit gathered by the players; to do so would ignore the potential value gathered by network users who recieve a high quality of service. Thus we consider social welfare to include both firm profits and user welfare. We prove that for a large class of these games, the price of anarchy is a small constant (assuming pure equilibria exist). In other words, even worst-case selfish behavior does not significantly decrease social welfare. Furthermore, we show that pure equlibria always do exist when latencies are linear.

1.2.3 The Effect of Coalition Formation

The final model we consider addresses the issue of collusion in games. Nearly all work in algorithmic game theory uses Nash equilibrium as the solution concept, i.e., defines Nash equilibrium as the outcome in a competitive game. The Nash equilibrium concept is based on the assumption that every player acts independently, in a solely self-interested fashion. And yet, we know that in reality, this tends not to be the case. In many competitive situations, the participating agents *do* collude: contracts are made, businesses agree to cooperate, and corporations merge. Just as we must acknowledge that agents may not all cooperate for the greater good, we must also recognize that, in many cases, agents will form small coalitions to improve their collective well-being.

In chapter 6, we present a formal framework to model *static* collusion in competitive games. We consider these coalitions in the context of a simple symmetric load-balancing game, in which players assign jobs to machines with load dependant latencies. We show the surprising result that when players collude in small groups, the price of anarchy can actually increase. In other words, the quality of the equilibria reached through limited coordination can in fact be worse than those reached in an entirely decentralized fashion.

We use the *price of collusion* to denote the factor by which the price of anarchy can increase when coalitions form in a game. The main result of this chapter is a proof that the price of collusion for our simple load-balancing game is a small constant when latencies are either convex or concave. For the case of convex latencies, we also prove that pure equilibria exist even after coalitions are formed. We do not know this to be true for concave latencies, and thus for these latencies, we also consider mixed equilibria and prove a slightly weaker bound on the price of collusion.

Chapter 2

Background and Related Work

2.1 Game Theory

In the section we review the basic concepts and definitions of game theory that are required to read this thesis. For a more thorough treatment of this matrial, see Fudenberg and Tirole [28].

2.1.1 Games and Strategies

A *strategic-form game*, henceforth refered to simply as a *game*, contains a set of *players* $\{1, \ldots, k\}$, and a strategy space $\mathcal{S}_i$ for each player i. We will sometimes refer to players as *agents*. We say an element $S_i \in \mathcal{S}_i$ is a *pure strategy for player* i, and we call a k-tuple $S = (S_1, \ldots S_k)$ consisting of one pure strategy for each player a *pure strategy*. A strategy can be thought of as specifying an action for each player.

To completely specify a game, we must decide how players evaluate strategies. Thus for each player i, we have a payoff function u_i, mapping pure strategies to real values. The value of $u_i(S)$ represents the utility player i recieves when each player selects a strategy as specified by S. Each player seeks to select a strategy so as to maximize his utility, given the strategies choosen by the other players. We assume that the structure of the game is common knowledge to all players.

For a concrete example, consider the following well known game, called *The Prisoners' Dilemma*. Two crooks are caught leaving the scene of the crime and are brought to the police station, where they are kept in separate cells. Both must choose between two strategies; each crook can either cooperate and say nothing, or can defect, turning

Table 2.1: The Prisoners' Dilemma.

	C	D
C	(-1,-1)	(-5,0)
D	(0,-5)	(-4,-4)

his partner in. If both crooks cooperate and remain silent, the minimal evidence they left behind is only enough to send them to jail for a single year each. If the crooks defect and turn each other in, they will both be sent to jail for four years. Finally, if one crook cooperates but his partner defects, the police will happily close the case, sending the cooperative crook to jail for five years, and letting the defector walk free.

This is naturally modeled as a game in which the crooks are the players. Both players have two strategies available to them; to cooperate (C), or to defect (D). The payoff for a player and a given strategy is the negative of the number of years he would spend in jail given the strategies selected. For example, $u_1(C, D) = -5$, since player 1 will go to jail for five years if he cooperates while his partner defects. We can represent this game easily as a 2-dimensional matrix, as shown in Table 2.1.

In this table, rows correspond to the strategy selected by the first player, and columns to the second player. The entry at each coordinate indicates the utility recieved by the first and second player respectively for that particular pure strategy.

Note that unlike the simple example of the Prisoners' Dilemma, for many of the games we consider in this thesis, the game description (the strategy spaces $\mathcal{S}_i$ and the payoff functions $u_i(S)$) will be given implicitly. For example, we will consider games in which player strategies may correspond to subsets of edges in some graph. We also consider games in which player strategies contain any non-negative real value, in which case an explicit game description would not even be finite.

2.1.2 Nash Equilibria

A *pure Nash equilibrium* is a pure strategy from which no player can unilaterally deviate and thereby improve his utility. More precisely, a pure strategy $S = (S_1, \ldots, S_k)$ is a pure Nash equilibrium if for any player i and for any other strategy $S'_i \in \mathcal{S}_i$ for player i,

$$u_i(S_1, \ldots, S_{i-1}, S'_i, S_{i+1}, \ldots, S_k) \leq u_i(S_1, \ldots, S_{i-1}, S_i, S_{i+1}, \ldots, S_k).$$

A pure Nash equilibrium is meant to represent the stable outcome of a game played by rational, self-interested agents. If we consider the Prisoners' Dilemma and apply the definition given above, we can easily verify that the only pure Nash equilibrium is (D, D). In other words, we should expect that if the crooks are rational and selfish, they will both turn each other in.

At first blush, this seems remarkably dumb; shouldn't they both cooperate and spend one year each in prison, instead of four? However, consider the perspective of one of the crooks. Suppose he knows that his partner is going to defect and turn him in. He can choose to do likewise and spend four years in jail, or cooperate, in which case he will have five years in the pen. Clearly the former is preferable. Now suppose that instead, he knows his partner will cooperate. In this case, his choice is between one and zero years in prison, but again, turining his partner in minimizes his time in the clink. Thus the only stable outcome is for both players to defect.

We now consider another very simple game, known as Matching Pennies. Here we have two players, each of whom has a coin that they can turn to be heads up (H) or tails up (T). Player 1 "wins" (i.e. recieves some utility) if the two coins match, and player 2 wins if they do not. As before, we can describe such a game as shown in Table 2.2.

While neither an exciting nor particularly interesting game, it brings up a key point; pure Nash equilibrium need not exist. Clearly there are only four pure strategies, and it

Table 2.2: Matching Pennies.

	H	T
H	(1,0)	(0,1)
T	(0,1)	(1,0)

is equally obvious that none of these are stable. Every pure strategy has a player who loses, and this player clearly has an incentive to flip his coin.

However, it turns out if we allow players to randomize over their stratgies, then stable solutions do exist. A *mixed strategy* A_i *for player* i is a random distribution over pure strategies in player i's strategy space. A *mixed strategy* $A = (A_1, \ldots, A_k)$ consists of a mixed strategy for each player. In a mixed strategy, there is a distribution over outcomes, and thus players are interested in their expected utility. More formally, we can define

$$u_i(A) = \sum_S u_i(S) \cdot \prod_j \text{Pr}[A_j \text{ chooses strategy } S_j].$$

A *mixed Nash equilibrium* is defined exactly as a pure Nash equilibrium, with mixed strategies in place of pure ones.

For example, consider the Matching Pennies game and suppose player 1 randomly chooses H and T with equal probability. Then for player 2, the expected utility for selecting both H and T is 1/2. Thus if both players select both heads and tails with probability 1/2, then neither can change to another mixed strategy and gather a greater utility.

Nash's much celebrated theorem states that such a mixed equilibrium always exists. The following version of this theorem is not the most general, but will be sufficient for the purposes of this thesis.

Theorem 2.1.1 (Nash 1951) *In any game with a finite number of players, a finite strategy space for each player, and real-valued payoff functions has a mixed Nash equilibrium.*

During the course of this thesis, we will prefer to consider pure equilibria, since in many settings, these more accurately reflect natural game play. However, in some cases pure equilibria are not guaranteed to exist, and thus we will extend our results to mixed equilibria when possible.

2.1.3 Evaluating the Outcome of a Game

The primary goal of this thesis is to understand the cost of selfish behavior. In particular, we want to compare the quality of solutions (i.e. strategies) reached in an uncoordinated and self-interested manner to the quality of those that can be reached by a centralized designer. To do this, we need to discuss what we mean by quality.

We take the quality of a solution to be its social welfare. We will specify precisely what we mean by this for any particular game we consider. However, generally we will view the social welfare of a strategy as the total (or equivalently, average) utility recieved by each player. Formally, we say that the social welfare of a strategy S is

$$c(S) = \sum_i u_i(S).$$

Similarly, we define the social welfare of a mixed strategy to be expected total player utility.

In this thesis, the one exception to this definition of social welfare appears in chapter 5. Here we consider a game in which players charge network users for service. We extend the definition of social welfare of a strategy to include both the total player utility and the total utility that users (who themselves are not explicitly modeled as players) gather by recieving service. We note that there are other reasonable definitions of solution quality, such as the egaltarian measure of minimum player utility. For the purposes of this thesis however, we will only consider social welfare.

We now have the tools to formally compare centralized solutions to decentralized solutions. For any given game, define OPT to be a strategy that maximizes social welfare. Typically, such a solution is not a Nash equilibrium. We are interested in comparing the cost of this solution to the cost of pure Nash equilibria.

Unfortunately, there may be multiple stable outcomes of a given game, possibly with different costs. Therefore, we will define BN to be a pure Nash equilibrium of maximum social value, and WN to be a pure Nash equilibrium of minimum social value. We define the *price of anarchy* [45][1] of the given game to be

$$PA = \frac{c(OPT)}{c(WN)}.$$

Similarly, we define the *price of stability* [7] to be

$$PS = \frac{c(OPT)}{c(BN)}.$$

When the price of anarchy of a particular game is small, we have the following very strong guarantee; any stable solution reached by rational, selfish agents is nearly as good as a solution that could be designed by a single, central authority. When possible, we would like to prove that the price of anarchy of a given game is small.

Often, such a result is not possible. However, even if the price of anarchy is large, the price of stability may still be small. In such a case, a central authority would be in the position to propose to the players a nearly optimal solution from which no player would have any incentive to deviate. Considering the price of stability can thus be viewed as a

[1]Koutsoupias and Papadimitriou define the price of anarchy in terms of mixed equilibria, but for most of this thesis, we will focus on pure equilibria.

compromise between considering fully cooperative solutions and those that are reached in an entirely uncoordinated fashion.

For convenience, when the games we consider involve costs (i.e. when utilities are negative), we will invert the definitions of price of anarchy and price of stability. Thus, these values are always at least 1. Also, as mentioned above, we will typically restrict ourselves to pure Nash equilbria, but the analogous definitions for the price of anarchy and the price of stability can be applied to mixed equilibria as well.

2.2 Related Work

In this section, we discuss previous work that is relevant to a significant portion of this thesis. We review the research that has particular relevance to specific chapters in the corresponding introductions.

The price of anarchy was introduced by Koutsoupias and Papadimitriou in [45, 54] and first applied to bound the cost of selfish behavior in a simple load-balancing game. In this game, each player controls a single weighted job. Each job must be processed on one of a number of machines. Each machine i has a load-dependent latency function, indicating the delay experienced by a user who schedules his job on i, given the total weight of all jobs scheduled for processing on i. Latencies are assumed to be linear and increasing. To measure the quality of a given solution, the authors consider the maximum delay experienced by any player. For stable solutions, they consider general mixed equilibria, i.e., solutions in which players may randomize over the set of machines on which to schedule their job. The primary focus of this paper is to bound the price of anarchy. The particular bounds they present depend on the number of machines, as well as whether or not the machines involved have the same latency functions.

Roughgarden and Tardos [63, 61] consider the price of anarchy in a similar game in which players route flow through a general network and attempt to minimize the latency they experience. In contrast to [45], they considered a nonatomic game, meaning that instead of k players, there is a continuum of players routing flow, each of whom individually controls a negligible volume of traffic. In this model there is no distinction between randomized and pure equilibria. In evaluating a solution [63, 61] consider social welfare (the sum of user delays), while [45] consider the egalitarian objective function of the worst delay.

Kothari, Suri, Tóth and Zhou [66, 44] were the first to focus on the social welfare (average delay) and pure Nash equilibria in a discrete load-balacing game. The social welfare of Nash equilibria in discrete routing and load-balancing games were further explored by Christodoulou and Koutsoupias [15, 16] and Awerbuch, Azar and Epstein [9], who extended the results to weighted games and mixed equilibria.

The nonatomic and unit job versions of the discrete routing and load balancing games are all special cases of congestion games. In a congestion game, the cost of using a machine or an edge depends on its load. Congestion games were introduced by Rosenthal [58] as a broad class of games possessing pure equilibria. Rosenthal proved that any congestion game is an *exact potential game*, a game where selfish moves are local steps that decrease a potential function Φ.

More formally, a game is an exact potential game if there exists a function Φ, from strategies to real values, such that for any strategies $\mathcal{A}$ and $\mathcal{A}'$ that only differ in the i^{th} coordinate, $\Phi(\mathcal{A}) - \Phi(\mathcal{A}') = u_i(\mathcal{A}) - u_i(\mathcal{A}')$. In other words, when any player changes their strategy, the change in Φ tracks the change in their cost exactly. For finite games, the existence of such a function guarantees a pure Nash equilibrium (e.g., the minimum of Φ). Later, Monderer and Shapley [52] proved the converse; every potential game is equivalent to a congestion game. Fabrikant et al. [24] study the complexity of computing Nash equilibria in certain classes of congestion games.

The bulk of the work in this thesis is based on four papers, the first three of which have appeared in conference proceedings. Chapter 3 is based on joint work with Anshelevich, Dasgupta, and Tardos [8]. Chapter 4 is based on joint work with Anshelevich, Dasgupta, Kleinberg, Tardos and Roughgarden [7]. Chapters 5 and 6 are both based on joint work with Hayrapetyan and Tardos [32, 33].

Chapter 3

Network Creation with Arbitrary Cost Sharing

3.1 Introduction

In many network settings, the system behavior arises from the actions of a large number of independent agents, each motivated by self-interest and optimizing an individual objective function. As a result, the global performance of the system may not be as good as in a case where a central authority can simply dictate a solution; rather, we need to understand the quality of solutions that are consistent with self-interested behavior. Recent theoretical work has framed this type of question in the following general form: how much worse is the solution quality at a Nash equilibrium[1], relative to the quality at a centrally enforced optimum? Questions of this genre have received considerable attention in recent years, for problems including routing [61, 63, 17], load balancing [20, 21, 45, 60], and facility location [67].

In this chapter, we introduce a game that models competitive network creation, and study the price of stability for this game. We consider a simple network design game where every agent has a specific connectivity requirement, i.e. each agent has a set of terminals and wants to build a network in which his terminals are connected. Each edge that can be built in the network has a cost, and each agent's goal is to pay as little as possible. When multiple players use a particular edge, they can share the cost of that edge, and thus there is an incentive for players to cooperate.

More precisely, we study the following network game, which we call the *connection game with arbitrary cost sharing*. For each game instance, we are given an undirected

[1]Recall that a Nash equilibrium is a state of the system in which no agent has an interest in unilaterally changing its own behavior.

graph G with non-negative edge costs. Let k denote the number of players in a game. Players form a network by purchasing some subgraph of G. Each player has a set of specified terminal nodes that he would like to see connected in the purchased network. With this as their goal, players offer payments indicating how much they will contribute towards the purchase of each edge in G. If the players' payments for a particular edge e sum to at least the cost of e, then the edge is considered *bought*, which means that e is added to our network and can now be used by any player.

Each player would like to minimize his total payments, but insists on connecting all of his terminals. We allow the cost of any edge to be shared by multiple players. Furthermore, once an edge is purchased, any player can use it to satisfy his connectivity requirement, even if that player contributed nothing to the cost of this edge. Finding the centralized optimum of the connection game, i.e. the network of bought edges which minimizes the sum of the players' contributions, is the classic network design problem of the generalized Steiner tree [4, 29].

Our primary focus will be on the price of stability for pure equilibria in this game. In addition to the price of stability, we will also consider the price of anarchy, approximate equilibria, and the existence of pure equilibria. In a game theoretic context it might seem natural to also consider *mixed* Nash equilibria, in which agents can randomly choose between different strategies. However, since we are modeling the construction of large-scale networks, randomizing over strategies is not a realistic option for players.

3.1.1 Results

We begin in section 3.2 with a formal definition of our game. We prove some basic properties of Nash equilibria for this game that are used throughout this chapter. An example for which pure equilibria do not exist is presented, and a simple example is given that demonstrates that the price of anarchy can be as large as the number of players, even for very simple networks.

In section 3.3 we consider the special case when the goal of each player is to connect a single terminal to a common source. We prove that in this case, there is a Nash equilibrium, the cost of which is equal to the cost of the optimal network. In other words, with a single source and one terminal per player, the price of stability is 1. Furthermore, given an $\varepsilon > 0$ and an α-approximate solution to the optimal network, we show how to construct in polynomial time an $(1+\varepsilon)$-approximate Nash equilibrium (players only benefit by a factor of $(1+\varepsilon)$ in deviating) whose total cost is within a factor of α to the optimal network.

We then generalize these results in two ways. First, we extend the results to the case when the graph is directed and players seek to establish a directed path from their terminal to the common source. Note that problems in directed graphs are often significantly more complicated than their undirected counterparts [13, 25]. Second, we relax the requirement that players connect their terminals. Instead, we suppose that player i has some maximum cost $\max(i)$ that he is willing to pay, and would rather stay unconnected if his cost exceeds $\max(i)$.

In section 3.5 we consider the more general case in which players may want to connect more than two terminals, and do not necessarily share a single source node. As we show in section 3.2, such games may not have any pure Nash equilibrium. Even when deterministic Nash equilibria do exist, the costs of different equilibria may differ by as much as a factor of k, the number of players, and even the price of stability may be nearly k. However, in section 3.5 we prove that there is always a 3-approximate equilibrium that pays for the optimal network. Furthermore, we show how to construct in polynomial time a $(4.65+\varepsilon)$-approximate Nash equilibrium whose total cost is within a factor of 2 to the optimal network.

Lastly, in section 3.6 we show that determining whether or not a Nash equilibrium exists is NP-complete when the number of players is part of the input. We also give a lower bound on the degree of instability that necessarily arises if we insist on building an optimal network.

3.1.2 Related Work

Our game is a simple model of how different service providers build and maintain the Internet topology. We use a game theoretic version of network design problems considered in approximation algorithms [29]. Fabrikant et al [23] study a different network creation game. Network games similar to that of [23] have also been studied for modeling the creation and maintenance of social networks [10, 35]. In the network games considered in [10, 23, 35] each agent corresponds to a single node of the network, and agents can only buy edges adjacent to their nodes. This model of network creation seems extremely well suited for modeling the creation of social networks. However, in the context of communication networks like the Internet, agents are not directly associated with individual nodes, and can build or be responsible for more complex networks.

There are many situations where agents can and do benefit by sharing the costs of certain expensive links in a network. Unlike the model presented in [23], the game we consider allows such cost sharing. Various forms of cost sharing have been studied extensively in the literature (see e.g. [26, 36] and the references there). The bulk of this work tends to assume a fixed underlying set of edges. Jain and Vazirani [37] and Kent and Skorin-Kapov [43] consider cost-sharing for a single source network design game. Cost-sharing games assume that there is a central authority that designs and maintains the network, and decides appropriate cost-shares for each agent, depending on the graph and all other agents, via a complex algorithm. The goal is to design a mechanism where truth-telling is a dominant strategy for the agents, i.e. selfish agents do not find it in their interest to misreport their utility (in hopes of being included in the network at a lesser costs). The agents' only role is to report their utility for being included in the network. Jain and Vazirani give a truthful mechanism to share the cost of the minimum spanning tree, which is a 2-approximation for the Steiner tree problem. In contrast, in the game we consider there is no central authority designing the Steiner tree or cost shares. Rather, we study Nash equilibria of our game. Also, in our game, agents must offer payments for each edge of the tree (modeling the cooperation of selfish agents), while in a mechanism design framework, agents pay the mechanism for the service, and do not care what edge they contribute to.

3.2 Model and Preliminaries

The Connection Game We now formally define the connection game for k players. Let an undirected graph $G = (V, E)$ be given, with each edge e having a nonnegative cost $c(e)$. Each player i has a set of terminal nodes that he must connect. The terminals of different players do not have to be distinct. A strategy of a player is a payment function p_i, where $p_i(e)$ is how much player i is offering to contribute to the cost of edge e. Any edge e such that $\sum_i p_i(e) \geq c(e)$ is considered *bought*, and G_p denotes the graph of bought edges with the players offering payments $p = (p_1, \ldots, p_k)$. Since each player must connect his terminals, all of the player's terminals must be connected in G_p. However, each player tries to minimize his total payments, $\sum_{e \in E} p_i(e)$.

A Nash equilibrium of the connection game is a payment function p such that, if players offer payments p, no player has an incentive to deviate from his payments. This is equivalent to saying that if p_j for all $j \neq i$ are fixed, then p_i minimizes the payments of player i. A $(1+\varepsilon)$-approximate Nash equilibrium is a function p such that no player i could decrease his payments by more than a factor of $1+\varepsilon$ by deviating, i.e. by using a different payment function p_i'.

Some Properties of Nash Equilibria Here we present several useful properties of Nash equilibria in the Connection Game. Suppose we have a Nash equilibrium p, and let T^i be the smallest tree in G_p connecting all terminals of player i. It follows from the definitions that (1) G_p is a forest, (2) each player i only contributes to costs of edges on T^i, and (3) each edge is either paid for fully or not at all.

Property 1 holds because if there was a cycle in G_p, any player i paying for any edge of the cycle could stop paying for that edge and decrease his payments while his terminals would still remain connected in the new graph of bought edges. Similarly, Property 2 holds since if player i contributed to an edge e which is not in T^i, then he could take away his payment for e and decrease his total costs while all his terminals would still remain connected. Property 3 is true because if i was paying something for e such that $\sum_i p_i(e) > c_e$ or $c_e > \sum_i p_i(e) > 0$, then i could take away part of his payment for e and not change the graph of bought edges at all.

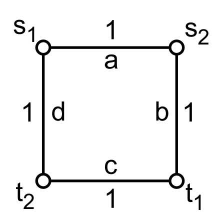

Figure 3.1: A game with no Nash equilibria.

Nash Equilibria May Not Exist It is not always the case that selfish agents can agree to pay for a network. There are instances of the connection game which have no deterministic Nash equilibria. In Figure 3.1, there are 2 players, one wishing to connect node s_1 to node t_1, and the other s_2 to t_2. Now suppose that there exists a Nash equilibrium p. By Property 1 above, in a Nash equilibrium G_p must be a forest, so assume without loss of generality it consists of the edges a, b, and c. By Property 2, player 1 only contributes to edges a and b, and player 2 only contributes to edges b and c. This means that edges a and c must be bought fully by players 1 and 2, respectively. At least one of the two players must contribute a positive amount to edge b. However, neither player can do that in a Nash equilibrium, since then he would have an incentive to switch to the strategy of only buying edge d and nothing else, which would connect his terminals with the player's total payments being only 1. Therefore, no Nash equilibria exist in this example.

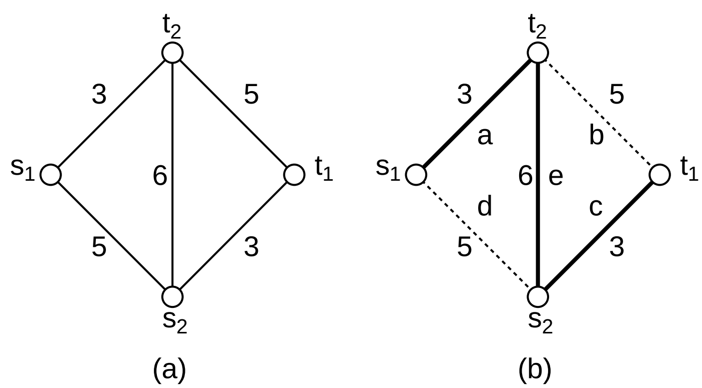

Figure 3.2: A game with only fractional Nash equilibria

Fractional Nash Equilibria When looking at the connection game, we might be tempted to assume that giving players the opportunity to share costs of edges is an unnecessary complication. However, sometimes players must share costs of edges for all players to agree on a network. There are game instances where the only Nash equilibria in existence require that players split the cost of an edge. We will call such Nash equilibria *fractional* and we will call Nash equilibria that do not involve players sharing costs of edges *non-fractional*.

Figure 3.2(a) is an example of a connection game instance where the only Nash equilibria are fractional ones. Once again, player 1 would like to connect s_1 and t_1, and player 2 would like to connect s_2 and t_2. First, note that there is a fractional Nash equilibrium, as shown in Figure 3.2(b). In this case we have that player 2 contributes 5 to edge e and player 1 contributes 1 to e and 3 to both of a and c. It is easy to confirm that neither player has an incentive to deviate.

Now we must show that there are no non-fractional Nash equilibria in this example. Observe that if edge e is not bought, then we have a graph which is effectively equivalent to the graph in which we showed there to be no Nash equilibria at all. Therefore any non-fractional Nash equilibria must buy edge e. Given that edge e must be bought, it is clear that player 2 will only contribute to edge e. For a Nash equilibrium p to be non-fractional, this would mean that player 2 either buys edge e fully or buys nothing at all. Suppose player 2 buys e. The only response for which player 1 would not want to deviate would be to buy a and c. But then player 2 has an incentive to switch to either edge b or d. Now suppose player 2 does not buy e. Then the only response for which player 1 would not want to deviate would be to either buy a and b or buy c and d. Either way, player 2 does not succeed in joining his source to his sink, and thus has an incentive to buy an edge. Hence, there are no non-fractional Nash equilibria in this graph.

The Price of Anarchy We have now shown that Nash equilibria do not have to exist. However, when they exist, how bad can these Nash equilibria be? As mentioned above, the price of anarchy often refers to the ratio of the worst (most expensive) Nash equilibrium and the optimal centralized solution. In the connection game, the price of anarchy is at most k, the number of players. This is simply because if the worst Nash equilibrium p costs more than k times OPT, the cost of the optimal solution, then there must be a player whose payments in p are strictly more than OPT, so he could deviate by purchasing the entire optimal solution by himself, and connect his terminals with smaller payments than before. More importantly, there are cases when the price of anarchy actually equals k, so the above bound is tight. This is demonstrated with the example in Figure 3.3. Suppose there are k players, and G consists of nodes s and t which are joined by 2 disjoint paths, one of length 1 and and one of length k. Each player has a terminal at s and t. Then, the worst Nash equilibrium has each player contributing 1 to the long path, and has a cost of k. The optimal solution here has a cost of only 1, so the price of anarchy is k. Therefore, the price of anarchy could be very high in the connection game. However, notice that in this example the *best* Nash equilibrium (which is each player buying $\frac{1}{k}$ of the short path) has the same cost as the optimal centralized solution. We have now shown that the price of anarchy can be very large in the connection game, but the price of stability remains worth considering, since the above example shows that it can differ from the (conventional) price of anarchy by as much as a factor of k.

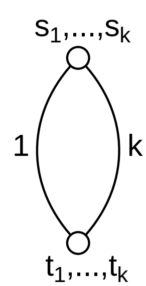

Figure 3.3: A game with price of anarchy of k

All the results in this section also hold if G is directed or if each player i has a maximum cost $\max(i)$ beyond which he would rather pay nothing and not connect his terminals.

3.3 Single Source Games

As we show in Section 3.6, determining whether or not Nash equilibria exist in a general instance of the connection game is NP-Hard. Furthermore, even when equilibria exist, they may be significantly more expensive than the centrally optimal network. In this section we define a class of games in which there is always a Nash equilibrium, and the price of stability is 1. Furthermore, we show how we can use an approximation to the centrally optimal network to construct a $(1+\epsilon)$-approximate Nash equilibrium in poly-time, for any $\epsilon > 0$.

Definition 3.3.1 *A* single source game *is a game in which all players share a common terminal s, and in addition, each player i has exactly one other terminal t_i.*

We will now show that the price of stability is 1 in single source games. To do this, we must argue that there is a Nash equilibrium that purchases T^*, the minimum cost Steiner tree on the players' terminal nodes. There are a number of standard cost-sharing methods for sharing the cost of a tree among the terminals. The two most commonly studied methods are the Shapley value and the Marginal Cost mechanisms [26]. The Marginal Cost (or VCG) mechanisms are very far from being budget balanced, i.e. the agents do not pay for even a constant fraction of the tree built. The Shapley value mechanism is budget balanced: the cost of each edge is evenly shared by the terminals that use the edge for their connection (i.e., the terminals in the subtree below the edge e). Unfortunately, this mechanism typically does not lead to a Nash equilibrium in our game: some players can have cheaper alternate paths, and hence benefit by deviating. However, in the next chapter, we will consider a variant of this game in which players are *required* to adhere to the Shapley value mechanism. We will show that by enforcing this “fair” form of cost sharing, many properties of the connection game change dramatically.

Another cost-sharing mechanism is presented by Jain and Vazirani [37]. This mechanism is both truthful and budget balanced, and pays for the minimum spanning tree, which is a 2-approximate budget balanced mechanism for the Steiner tree problem. However, it is only a 2-approximation, and the cost-shares are not associated with edges that the agents use. Here we will show that while the traditional Steiner tree cost-sharing methods do not lead to a Nash equilibrium, such a solution can be obtained.

Theorem 3.3.2 *In any single source game, there is a Nash equilibrium which purchases T^*, a minimum cost Steiner tree on all players' terminal nodes.*

Proof : Given T^*, we present an algorithm to construct payment strategies p. We will view T^* as being rooted at s. Let T_e be the subtree of T^* disconnected from s when e is removed. We will determine payments to edges by considering edges in reverse breadth first search order. We determine payments to the subtree T_e before we consider edge e. In selecting the payment of agent i to edge e we consider c', the cost that player i faces if he deviates in the final solution: edges f in the subtree T_e are considered to cost $p_i(f)$, edges f not in T^* cost $c(f)$, while all other edges cost 0. We never allow i to contribute so much to e that his total payments exceed his cost of connecting t_i to s.

Algorithm 3.4 `Initialize` $p_i(e) = 0$ `for all players` i `and edges` e.

`Loop through all edges` e `in` T^* `in reverse BFS order.`

  `Loop through all players` i `with` $t_i \in T_e$ `until` e `paid for.`

    `If` e `is a cut in` G `set` $p_i(e) = c(e)$.

    `Otherwise`

      `Define` $c'(f) = p_i(f)$ `for all` $f \in T^*$ `and`

        $c'(f) = c(f)$ `for all` $f \notin T^*$.

      `Define` χ_i `to be the cost of the cheapest path from` s `to`

```
      t_i in G \ {e} under modified costs c'.
    Define p_i(T*) = ∑_{f∈T*} p_i(f).
    Define p(e) = ∑_j p_j(e).
    Set p_i(e) = min{χ_i − p_i(T*), c(e) − p(e)}.
  end
 end
end
```

We first claim that if this algorithm terminates, the resulting payment forms a Nash equilibrium. Consider the algorithm at some stage where we are determining i's payment to e. The cost function c' is defined to reflect the costs player i faces if he deviates in the final solution. We never allow i to contribute so much to e that his total payments exceed his cost of connecting t_i to s. Therefore it is never in player i's interest to deviate. Since this is true for all players, p is a Nash equilibrium.

We will now prove that this algorithm succeeds in paying for T^*. In particular, we need to show that for any edge e, the players with terminals in T_e will be willing to pay for e. Assume the players are unwilling to buy an edge e. Then each player has some path which explains why it can't contribute more to e. We can use a carefully selected subset of these paths to modify T^*, forming a cheaper tree that spans all terminals and doesn't contain e. This would clearly contradict our assumption that T^* had minimum cost.

Define player i's *alternate path* A_i to be the path of cost χ_i found in Algorithm 3.4, as shown in Figure 3.4(a). If there is more than one such path, choose A_i to be the path which includes as many ancestors of t_i in T_e as possible before including edges outside of T^*. To show that all edges in T^* are paid for, we need the following technical lemma concerning the structure of alternate paths.

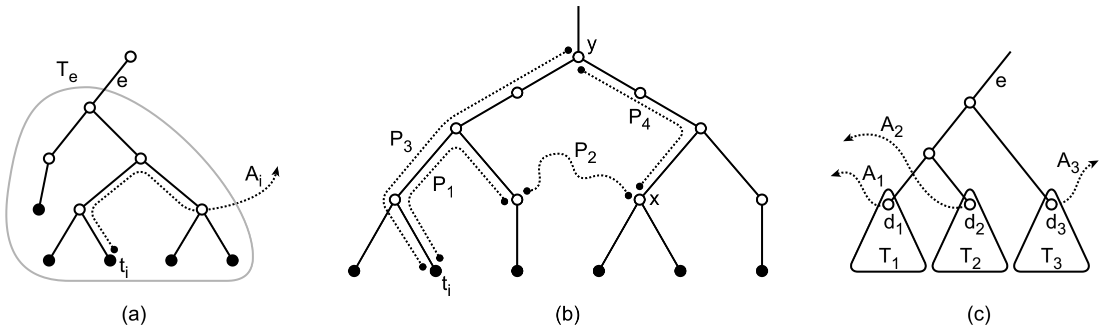

Figure 3.4: Alternate paths in single source games.

Lemma 3.4.1 *Suppose A_i is i's alternate path at some stage of the algorithm. Then there are two nodes v and w on A_i, such that all edges on A_i from t_i to v are in T_e, all edges between v and w are in $E \setminus T^*$, and all edges between w and s are in $T^* \setminus T_e$.*

Proof : Once A_i reaches a node w in $T^* \setminus T_e$, all subsequent nodes of A_i will be in $T^* \setminus T_e$, as all edges f in $T^* \setminus T_e$ have cost $c'(f) = 0$ and the source s is in $T^* \setminus T_e$. Thus, suppose A_i begins with a path P_1 in T_e, followed by a path P_2 containing only edges not in T^*, before reaching a node x in T_e, as shown in Figure 3.4(b). Let y be the lowest common ancestor of x and t_i in T_e. Observe that P_1 is strictly below y. Define P_3 to be the path from t_i to y in T_e, and define P_4 to be the path from y to x in T_e. We now show that under the modified cost function c', $P_3 \cup P_4$ is at least as cheap as $P_1 \cup P_2$. Since $P_1 \cup P_2$ includes a higher ancestor of t_i than A_i (namely y), this contradicts our choice of A_i.

Consider the iterations of the algorithm during which player i could have contributed to edges in P_3. At each of these steps the algorithm computes a cheapest path from t_i to s. At any time, player i's payments are upper bounded by the modified cost of his alternate path, which is in turn upper bounded by the modified cost of any path, in particular A_i. Furthermore, at each of these steps the modified costs of all edges in A_i above x are 0. Therefore i's contribution to P_3 is always at most the modified cost of

$P_1 \cup P_2$. The modified cost of P_4 is always 0, as none of the edges in P_4 are on player i's path from t_i to s in T^*. Together these imply that $c'(P_3 \cup P_4) = c'(P_3) \leq c'(P_1 \cup P_2)$. ∎

Thus, players' alternate paths may initially use some edges in T_e, but subsequently will exclusively use edges outside of T_e. We use this fact in the following lemma.

Lemma 3.4.2 *Algorithm 3.4 fully pays for every edge in T^*.*

Proof : Suppose that for some edge e, after all players have contributed to e, $p(e) < c(e)$.

For each player i, consider the longest subpath of A_i containing t_i and only edges in T_e. Call the highest ancestor of t_i on this subpath i's *deviation point*, denoted d_i. Note that it is possible that $d_i = t_i$. Let D be a minimum set of deviation points such that every terminal in T_e has an ancestor in D.

Suppose every player i with a terminal t_i in D deviates to A_i, as shown in Figure 3.4(c), paying his modified costs to each edge. Any player i deviating in this manner does not increase his total expenditure, as player i raised $p_i(e)$ until p_i matched the modified cost of A_i. The remaining players leave their payments unchanged.

We claim that now the edges bought by players with terminals in T_e connect all these players to $T^* \setminus T_e$. To see this, first consider any edge f below a deviation point d_i in D. By Lemma 3.4.1, player i is the only deviating player who could have been contributing to f. If i did contribute to f, then f must be on the unique path from t_i to d_i in T_e. But by the definition of d_i, this means that f is in A_i. Thus player i will not change his payment to f.

Define T_i to be the subtree of T_e rooted at d_i. We have shown that all edges in T_i have been bought. By Lemma 3.4.1, we know that A_i consists of edges in T_i followed by edges in $E \setminus T$ followed by edges in $T^* \setminus T_e$. By the definition of c', the modified cost of those edges in $E \setminus T^*$ is their actual cost. Thus i pays fully for a path connecting T_i to $T^* \setminus T_e$.

We have assumed that the payments generated by the algorithm for players with terminals in T_e were not sufficient to pay for those terminals to connect to $T^* \setminus T_e$. However, without increasing any players' payments, we have managed to buy a subset of edges which connects all terminals in T_e to $T^* \setminus T_e$. This contradicts the optimality of T^*. Thus the algorithm runs to completion. ■

Since we have also shown that the algorithm always produces a Nash equilibrium, this concludes the proof of the theorem. ■

We will now argue that Algorithm 3.4 works even if the graph is directed. It is still the case that if the algorithm does succeed in assigning payments to all edges, then we are done. Hence, to prove correctness, we will again need only show that failure to pay for an edge implies the existence of a cheaper tree, thus yielding a contradiction. The problem is that Lemma 3.4.1 no longer holds; it is possible that some of the players attempting to purchase an edge e have an alternate paths which repeatedly moves in and out of the subtree T_e. Thus, the argument is slightly more complex.

Theorem 3.4.3 *Algorithm 3.4 fully pays for every edge in T^* for directed graphs, and thus the price of stability for single source directed games is also 1.*

Proof : Suppose the algorithm fails to pay for some edge e. At this point, every player i with a terminal in T_e has an alternate path A_i, as defined earlier. Define D to be the set of vertices contained in both T_e and at least one alternate path. Note that D contains all terminals that appear in T_e. We now create $D' \subseteq D$ by selecting the *highest* elements of D; we select the set of nodes from D that do not have ancestors with respect to T_e in D. Every terminal in T_e has a unique ancestor in D' with respect to T_e, and every node in

D' can be associated with at least one alternate path.

For any node $v_j \in D'$, let A_j be the alternate path containing v_j. If more than one such path exists, simply select one of them. Define A'_j to be the portion of this path from v_j to the first node in $T \setminus T_e$. We can now form T' as the union of edges from $T \setminus T_e$, all paths A'_j, and every subtree of T_e rooted at a node in D'. T' might not be a tree, but breaking any cycles yields a tree which is only cheaper.

It is clear that all terminals are connected to the root in T', since every terminal in T_e is connected to some node in D', which in turn is connected to $T \setminus T_e$. Now we just need to prove that the cost of our new tree is less than the cost of the original. To do so, we will show that the total cost of the subtrees below nodes in D', together with the cost of adding any additional edges needed by the paths A'_j, is no greater than the total payments assigned by the algorithm to the players in T_e thus far. Hence it will be helpful if we continue to view the new tree as being paid for by the players. In particular, we will assume that all players maintain their original payments for all edges below nodes in D', and the additional cost of building any path A'_j is covered by the player for which A_j was an alternate path. It now suffices to show that no player increases their payment.

For the case of those players who are not associated with a node from D', this trivially holds, since their new payments are just a subset of their original payments. Now consider a player i who must pay for any unbought edges in the path A'_j, which starts from node $v_j \in D'$. Note that player i's terminal might not be contained within the subtree rooted at v_j. If it is, then we are done, since in this case, player i's new cost is at most the cost of A_j, which is exactly i's current payment.

Thus suppose instead that player i's terminal lies in a subtree rooted at a node $v_k \in D'$. As shown in figure 3.5, define u to be the least common ancestor of v_k and v_j in T_e. Observe that u can not be either v_k or v_j, as this would contradict the minimality of the

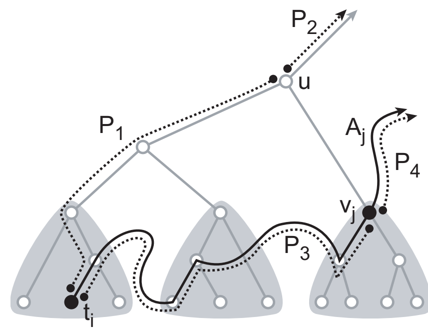

Figure 3.5: Alternate paths in a directed single source game.

set D'. Define P_1 to be the current payments made by player i from its terminal to u, and let P_2 be the current payments made by player i from u to e (inclusive). Define P_3 as the cost of $A_j \setminus A'_j$ and let P_4 be the cost of A'_j. By the definition of alternate path,

$$P_1 + P_2 = P_3 + P_4.$$

Furthermore, since we have already successfully paid for a connection to u, we know that

$$P_3 \geq P_1,$$

since otherwise, when we were paying for the edges between v_k and u, player i would have had an incentive to deviate by purchasing P_3 and then using the path from v_j to u in T_e, which would have been free for i. Hence $P_4 \leq P_2$.

Therefore we can bound player i's contribution to edges below D' by P_1 (since u lies above v_k), and we can bound player i's contribution to A'_j by P_2. Taken together, we have that player i's new cost has not increased. Thus no player has increased their payments, and yet the total cost of the tree has decreased, which is a contradiction. ■

We have shown that the price of stability in a single source game is 1. However, the algorithm for finding an optimal Nash equilibrium requires us to have a minimum cost Steiner tree on hand. Since this is often computationally infeasible, we present the following result.

Theorem 3.4.4 *Suppose we have a single source game and an α-approximate minimum cost Steiner tree T. Then for any $\varepsilon > 0$, there is a poly-time algorithm which returns a $(1+\varepsilon)$-approximate Nash equilibrium on a Steiner tree T', where $c(T') \leq c(T)$.*

Proof : To find a $(1+\varepsilon)$-approximate Nash equilibrium, we start by defining $\gamma = \frac{\varepsilon c(T)}{(1+\varepsilon)n\alpha}$. We now use Algorithm 3.4 to attempt to pay for all but γ of each edge in T. Since T is not optimal, it is possible that even with the γ reduction in price, there will be some edge e that the players are unwilling to pay for. If this happens, the proof of Theorem 3.3.2 indicates how we can rearrange T to decrease its cost. If we modify T in this manner, it is easy to show that we have decreased its cost by at least γ. At this point we simply start over with the new tree and attempt to pay for that.

Each call to Algorithm 3.4 can be made to run in polynomial time. Furthermore, since each call which fails to pay for the tree decreases the cost of the tree by γ, we can have at most $\frac{(1+\varepsilon)\alpha n}{\varepsilon}$ calls. Therefore in time polynomial in n, α an ε^{-1}, we have formed a tree T' with $c(T') \leq c(T)$ such that the players are willing to buy T' if the edges in T' have their costs decreased by γ.

For all players and for each edge e in T', we now increase $p_i(e)$ in proportion to p_i so that e is fully paid for. Now T' is clearly paid for. To see that this is a $(1+\varepsilon)$-approximate Nash equilibrium, note that player i did not want to deviate before his payments were increased. If we let m' be the number of edges in T', then i's payments were increased by

$$\gamma\frac{p_i(T')}{c(T') - m'\gamma}m' = \frac{\varepsilon c(T)p_i(T')m'}{(1+\varepsilon)n\alpha(c(T') - m'\gamma)} \leq \frac{\varepsilon c(T)p_i(T')}{\alpha(1+\varepsilon)(1-\varepsilon)c(T')} \leq \varepsilon p_i(T').$$

Thus any deviation decreases a players payment by at most an ε factor of the cost of his current strategy. ■

Modeling a Maximum Allowable Cost

We note here that since our theorems apply in the directed case, we can extend our model and give each player i a maximum cost $\max(i)$ beyond which he would rather pay nothing and not connect his terminals. It suffices to make a new terminal t'_i for each player i, with a directed edge of cost 0 to t_i and a directed edge of cost $\max(i)$ to s. Note that alternatively, we can simply adapt the proof used in the undirected case to account for a maximum allowable cost without any significant conceptual changes.

3.5 General Connection Games

In this section we deal with the general case of players that can have different numbers of terminals and do not necessarily share the same source terminal. As stated before, in this case the price of anarchy can be as large as k, the number of players, and pure equilibria may not exist at all. Furthermore, it turns out that even when pure stable solutions do exist, the price of stability may be quite large in this general setting.

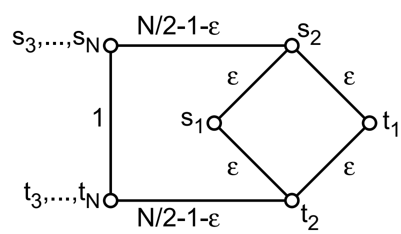

Figure 3.6: A game with high price of stability.

Consider the graph illustrated in Figure 3.6, where each player i owns terminals s_i and t_i. The optimal centralized solution has cost $1 + 3\varepsilon$. If the path of length 1 were bought, each player $i > 2$ will not want to pay for any ε edges, and therefore the situation of players 1 and 2 reduces to the example in Section 3.2 of a game with no Nash equilibria. Therefore, any Nash equilibrium must involve the purchase of the path of length $k - 2$. In fact, if each player $i > 2$ buys $\frac{1}{k-2}$ of this path, then we have a Nash equilibrium. Therefore, for any $k > 2$, there exists a game with the price of stability being nearly $k - 2$.

Because the price of stability can be as large as $\Theta(k)$, and sometimes pure Nash equilibria may not exist at all, we cannot hope to be able to provide cheap Nash equilibria for the multi-source case. Therefore, we consider how cheap α-approximate Nash equilibria with small α can be, and obtain the following result, which tells us that there always exists a 3-approximate Nash equilibrium as cheap as the optimal centralized solution.

Theorem 3.5.1 *For any optimal centralized solution T^*, there exists a 3-approximate Nash equilibrium such that the purchased edges are exactly T^*.*

To prove this, we use Theorem 3.5.2, which provides sufficient conditions for an approximate Nash equilibrium. In Subsection 3.5.2 we complete the proof of Theorem 3.5.1. In Subsection 3.5.3 we address the very special case where the underlying graph is a path, and in Subsection 3.5.4 we give lower bounds and a polynomial time algorithm for finding an approximate Nash equilibrium.

3.5.1 Sufficient Conditions for Approximate Nash Equilibria

Given a set of bought edges T, denote by a *stable payment* p_i for player i a payment such that player i has no better deviation than p_i, assuming that the rest of T is bought by the other players. A Nash equilibrium must consist of stable payments for all players.

However, what if in some solution, a player's payment p_i is not stable, but is a union of a small number of stable payments? This implies that each player's best deviation is not much less than its current payment. Specifically, we have the following general theorem.

Theorem 3.5.2 *Suppose we are given a payment scheme $p = (p_1, \ldots, p_k)$, with the set of bought edges T. If for all i, the payment p_i is a union of at most α stable payments (with respect to T), then p is an α-approximate Nash equilibrium.*

Proof : Let p'_i be the best deviation of player i given p, and let $p_i^1, p_i^2, \ldots, p_i^\alpha$ be the stable payments which together form p_i. The fact that p'_i is a valid deviation for i means that the set of bought edges T with p_i taken out and p'_i added still connects the terminals of i. p_i^j being a stable payment means that if i only pays for p_i^j and the rest of T is bought by other players, then the best deviation of i is al least as expensive than p_i^j. In this case, p'_i is still a possible deviation, since if taking out p_i and adding p'_i connects the terminals of i, then so does taking out p_i^j and adding p'_i. Therefore, we know that the cost of p'_i is no smaller than the cost of any p_i^j, and $\alpha \cdot cost(p'_i) \geq cost(p_i)$, as desired. ■

Notice that the converse of this theorem is not true. Consider an example where player i is contributing to an edge which it does not use to connect its terminals. If this edge is cheap, this would still form an approximate Nash equilibrium. However, this edge would not be contained in any stable payment of player i, so p_i would not be a union of stable payments.

To prove Theorem 3.5.1, we will construct a payment scheme on the optimal centralized solution such that each player's payment is a union of 3 stable payments. The stable payments we use for this purpose involve each edge being paid for by a single player, and have special structure. We call these payments *connection sets*. Since there is no sharing of edge costs by multiple players in connection sets, we will often use sets of edges and sets of payments interchangeably. T^* below denotes an optimal centralized solution, which we know is a forest.

Definition 3.5.3 *A connection set S of player i is a subset of edges of T^* such that for each connected component C of the graph $T^* \setminus S$, we have that either*
(1) any player that has terminals in C has all of its terminals in C, or
(2) player i has a terminal in C.

Intuitively, a connection set S is a set such that if we removed it from T^* and then somehow connected all the terminals of i, then all the terminals of all players are still connected in the resulting graph. We now have the following lemma, the proof of which follows directly from the definition of a connection set.

Lemma 3.5.4 *A connection set S of player i is a stable payment of i with respect to T^*.*

Proof : Suppose that player i only pays exactly for the edges of S, and the other players buy the edges in $T^* \backslash S$. Let Q be a best deviation of i in this case. In other words, let Q be a cheapest set of edges such that the set $(T^* \backslash S) \cup Q$ connects all the terminals of i. To prove that S is a stable payment for i, we need to show that $cost(S) \leq cost(Q)$.

Consider two arbitrary terminals of any player. If these terminals are in different components of $T^* \backslash S$, then by definition of connection set, each of these components must have a terminal of i. Therefore, all terminals of all players are connected in $(T^* \backslash S) \cup Q$, since $(T^* \backslash S) \cup Q$ connects all terminals of i. Since T^* is optimal, we know that $cost(T^*) \leq cost((T^* \backslash S) \cup Q)$. Since $S \subseteq T^*$ and Q is disjoint from $T^* \backslash S$, then $cost((T^* \backslash S) \cup Q) = cost(T^*) - cost(S) + cost(Q)$, and so $cost(S) \leq cost(Q)$. ■

More generally, if we assume player i is paying exactly for a set $S \subseteq T^*$, consider the components of $T^* \backslash S$ which do not obey Property 1 of Definition 3.5.3. We will call such components *uncoupled*, since there are terminals in them which are not connected. If S is a connection set, each such component contains a terminal of i. Call these terminals the *certificate terminals* of S. If S is not a connection set, there may be components without certificate terminals in them.

3.5.2 Proof of Theorem 3.5.1 (Existence of 3-Approx Nash Equilibrium)

In this subsection, we prove that for any optimal centralized solution T^*, there exists a 3-approximate Nash equilibrium such that the purchased edges are exactly T^*.

Let T^i be the unique smallest subtree of T^* which connects all terminals of player i. For simplicity of the proof, we assume that T^* is a tree, since otherwise we can apply this proof to each component of T^*.

By Theorem 3.5.2 and Lemma 3.5.4, to prove Theorem 3.5.1 it is enough to find a payment scheme for T^* so that no player pays for more than 3 connection sets. We now exhibit such a scheme. First, each player i pays for the edges belonging only to T^i and no other tree T^j. This is clearly a connection set, so we want each player to pay for at most 2 more. We can contract the edges now paid for, forming a new tree T^* which the players must pay for, and on which each edge belongs to at least two different T^i's.

Intuition and Outline of Proof By Lemma 3.5.5, we know how to make each player pay for at most two connection sets if T^* is just a path. The idea of the proof of Theorem 3.5.1 is to partition the tree T^* into paths, use Lemma 3.5.5 to form payments on these paths, and then "hook up" all these payments together so that in total, each player is still paying for 2 connection sets.

The partitioning process into paths works as follows. At first we take an arbitrary demand path R (i.e. take 2 terminals belonging to the same player, and take the path in T^* between them). Now we use Lemma 3.5.5 on this path. We get payments for this path, and a set A of terminals. We add the paths starting at terminals of A and ending at R to our partition, and continue like this recursively. (This is the purpose of the set A in Lemma 3.5.5).

Now we must prove that these payments which were done on a "path-by-path" basis form a "2-connection-set" payment on the entire tree T^*. This is shown in Lemma 3.5.6.

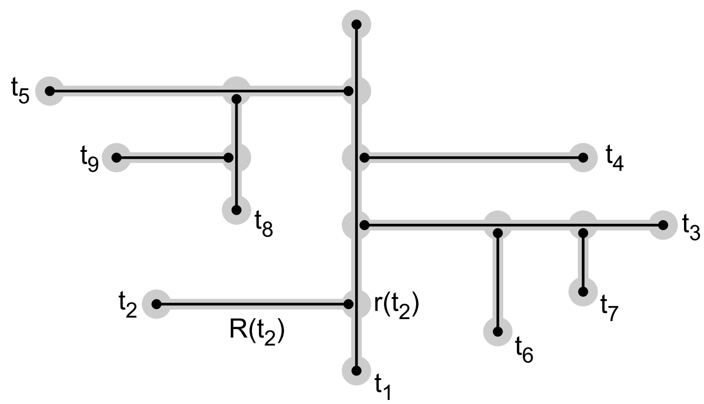

Figure 3.7: A decomposition of T^* into paths $R(t)$

The Payment Scheme and Path Partitioning Now we recursively assign players to the edges of T^*. Each edge will be assigned a player, which will pay for it. At every point in this process, we have a set of directed paths R to be paid for. Each of these paths is labeled with some terminal s at which it begins, and each of these paths ends at a node of a path paid for at a previous time. We will call such a path $R(s)$, since for each terminal there will be at most 1 path starting at that terminal, and we will call the last node of this path $r(s)$. Figure 3.7 shows a decomposition of T^* into these paths after this recursion is done. To explain how this process of path decomposition proceeds, we first need the following key lemma about connection sets in paths.

Suppose our entire graph is a path P, with nodes v_r, $r = 1 \dots n$. Denote the set of all terminals located at v_r by U_r. Select an arbitrary terminal $s \in U_1$ such that the player owning s also owns a terminal in U_n (if none such exists, things only become easier). The following technical result holds in this scenario.

Lemma 3.5.5 *If $U_1 \neq \{s\}$, then there exists a set of terminals $A = \{s\} \cup \{s_1, \dots, s_{n-1}\}$, $s_r \in U_r$, and a payment scheme on path P with each player i paying exactly for the set of edges S_i such that:*

1. *Any player i owning a terminal in U_n pays for at most 1 connection set, with all certificate terminals from A or U_n.*

2. *If player i owns s, it pays for at most 1 connection set, with all certificate terminals from A (not U_n).*

3. *If players i does not own a terminal in U_n, there is at most one uncoupled component of $P \backslash S_i$ without a certificate terminal of i from A.*

The proof of this lemma can be found in the next subsection. We can now explain the recursive process of path decomposition. Initially, select $R(s)$ to be a path from an arbitrary terminal s to another terminal of the same player in T^i, direct this path away from s, and set $R = \{R(s)\}$. At each time step t in the recursion, we have a subset T of T^* which has already been assigned payments, a set of paths R that needs to be paid for, and the rest of T^*, which has not been partitioned into paths yet. The recursion proceeds until all of T^* is paid for. For each $R(s) \in R$, we do the following:

1. Invoke Lemma 3.5.5, with $P = R(s)$ and U_r being all the terminals in the subtree of T^* containing v_r that we obtain by cutting the edges in $R(s)$ incident to v_r. $U_1 \neq \{s\}$ because we have contracted all edges belonging to a single T^i. From

Lemma 3.5.5, we obtain a payment assignment for $R(s)$, and a set of terminals $A = \{s_1, \ldots\}$ such that $s_r \in U_r$.

2. For each r, let $R(s_r)$ be the path from $s_r \in A$ to v_r in T^*, and add this path to the set of paths R that needs to be paid for.

3. For every edge e incident to a node $v_r \in R(s)$ which is not already paid for, and which is not covered by R, let s' be an arbitrary terminal in the subtree not containing v_r obtained by cutting e, with the condition that if player i owns s', then $e \in T^i$. Let $R(s')$ be the path from s' to v_r, and add it to R.

In this recursive manner, we decompose all of T^* into paths, and assign which player pays for which edges. Suppose this process runs for τ time steps (so T^* is decomposed into τ paths). Let S_i be the edges paid for by player i after this process, and let $S_i(t, t')$ be the edges assigned to player i in the time steps t and t', inclusive. In other words, $S_i = S_i(1, \tau)$.

To prove Theorem 3.5.1, we now need to prove that in this payment scheme, each player pays for at most 2 connection sets. In fact, we now prove the following lemma, which is a stronger statement.

Lemma 3.5.6 *Suppose the recursive process assigns edges S_i to be paid for by player i. Then there is at most 1 connected component of $T^* \backslash S_i$ which is uncoupled and does not contain a terminal of i. Furthermore, this component must intersect the path $R(s)$ which is the first path paid for sharing edges with T^i.*

Proof : For each path $R(s)$ in the above recursive process, let $T(s)$ be the tree containing $R(s)$ obtained by removing the last node $r(s)$ of $R(s)$ from T^*. Along with this lemma, we will also prove the following invariant: *If i owns s, then the component of $T^*\backslash S_i(t,\tau)$ containing $r(s)$ always contains a terminal of i in $T(s)$.*

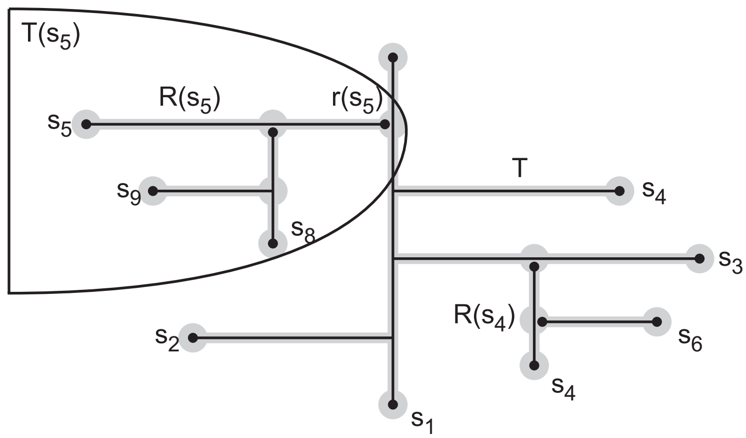

Figure 3.8: A snapshot of the recursive process at the point where $R(s_5)$ is being paid for.

We prove this inductively in reverse, i.e. at each step of the induction we show that the lemma holds for $S_i(t,\tau)$ instead of S_i, decreasing t at every step. In the base case of $S_i(\tau+1,\tau) = \emptyset$, the players have not been assigned any payments, so the lemma and the invariant are trivially true. Now assume that the lemma and the invariant hold for $S_i(t+1,\tau)$. We want to show that they also hold for $S_i(t,\tau)$. At step t in the above recursive process, there is some subtree T of T^* which has already been assigned payments, there is a set of paths R we are currently trying to pay for, and there is the rest of T^*, which has not been partitioned into paths yet. Figure 3.8 shows the tree T^* at the time step when $R(s_5)$ is being paid for. In the figure, R consists of $R(s_5)$ and $R(s_4)$, with the parts of T^* involving s_6, s_8, and s_9 not having been paid for or put into R.

At step t, we invoke Lemma 3.5.5 to assign player payments for a path $R(s) \in R$. Suppose $R(s)$ is of length n, and let P and U_r be the path and sets which we give to

Lemma 3.5.5 as inputs. We now have 3 cases to consider.

Case 1: Player i has no terminals outside $T(s)$. If all of i's terminals are in the same set U_r, then the Lemma still holds by the inductive hypothesis. Otherwise, every component of $T^*\backslash S_i(t,\tau)$ that does not intersect $R(s)$ is either coupled or has a terminal of i by the inductive hypothesis. By Lemma 3.5.5, we know that at time step t, all except 1 connected component of $T^*\backslash S_i(1,t)$ intersecting $R(s)$ is coupled or contains a certificate terminal of i. The recursive process at step t adds paths starting at these certificate terminals to R, and because of the invariant being true after step t, all the corresponding components of $T^*\backslash S_i$ still contain a terminal of i. [Maybe put in more explanation about what "corresponding" means, and that the coupled components of $T^*\backslash S_i(1,t)$ are still coupled in $T^*\backslash S_i(t,\tau)$] This proves the Lemma for this case. The fact that i has no terminals outside $T(s)$ implies that $R(s)$ is the first path being paid for that intersects T^i, as desired.

If i owns s, we must also show that the invariant still holds. In the recursive process, s can only be chosen as the terminal to head $R(s)$ if all of $R(s)$ is in T^i. Since i has no terminals outside $T(s)$, this implies that $r(s)$ is a terminal of i. Therefore, the invariant is satisfied.

Case 2: Player i has terminals outside $T(s)$ and owns s. As before, the components of $T^*\backslash S_i(t,\tau)$ not intersecting $R(s)$ must either be coupled or contain a terminal of i by the inductive hypothesis. When we invoke Lemma 3.5.5 at step t, U_n contains a terminal of i, so Lemma 3.5.5 tells us that i pays for at most 1 connection set $S_i(t,t)$ in $R(s)$, and all of the certificate terminals of $S_i(t,t)$ are from a set $A = \{s\} \cup \{s_1, \ldots, s_{n-1}\}$. This implies that every component of $T^*\backslash S_i(1,t)$ intersecting $R(s)$ is either coupled or contains a certificate terminal of i from A. By the same argument as in Step 1, the same must be true for all components of $T^*\backslash S_i(t,\tau)$.

We must also show that the invariant still holds. Consider the component C of $T^*\backslash S_i(t,\tau)$ containing $r(s)$. If it is coupled, then it must contain all terminals of i, since i has terminals outside $T(s)$. Therefore, it must contain s, since it is a terminal of i, and so the invariant is satisfied. If C is uncoupled, it must have contained a certificate terminal of i from A during the invocation of Lemma 3.5.5. That terminal s' is contained in $T(s)$, and a path $R(s')$ must have been added to R at step t. By the invariant, there must be a terminal of i in the component of $T(s')\backslash S_i(t+1,\tau)$ intersecting $R(s)$. Since $S_i(t,t)$ does not involve any payments to $T(s')$, this means the invariant still holds.

Case 3: Player i has terminals outside $T(s)$, but does not own s. This means when we invoke Lemma 3.5.5, U_n contains a terminal of i, so Lemma 3.5.5 tells us that i pays for at most 1 connection set $S_i(t,t)$ in $R(s)$, and all of the certificate terminals of $S_i(t,t)$ are from A or U_n. By an argument similar to Case 2, both the Lemma and the invariant still hold. ■

3.5.3 Proof of Lemma 3.5.5 (Connection Sets in Paths)

In this section, we prove Lemma 3.5.5 about connection sets in paths. This lemma is a basic building block for our proof of Theorem 3.5.1, and is a rather technical result.

Proof : Throughout this proof, we order the nodes of P from v_1 to v_n. For example, the "first" terminal of i will mean the one closest to v_1.

For every terminal $u \in U_r$, with $r \neq n$, define a subpath $Q(u)$ as follows. If there exists a node v_ℓ with $\ell > r$ and the player who owns u also owns a terminal in U_ℓ, then set $Q(u)$ to be the subpath of P from v_r to v_ℓ. If there is no such node, set $Q(u)$ to be the subpath of P starting at the first terminal of the player who owns u, and ending at v_r.

In the special case where the same player owns both u and s, if there exists a node

v_ℓ with $\ell < r$ and the player who owns u also owns a terminal in U_ℓ, then set $Q(u)$ to be the subpath from v_ℓ to v_r. If there is no such node v_ℓ, set $Q(u)$ to be empty.

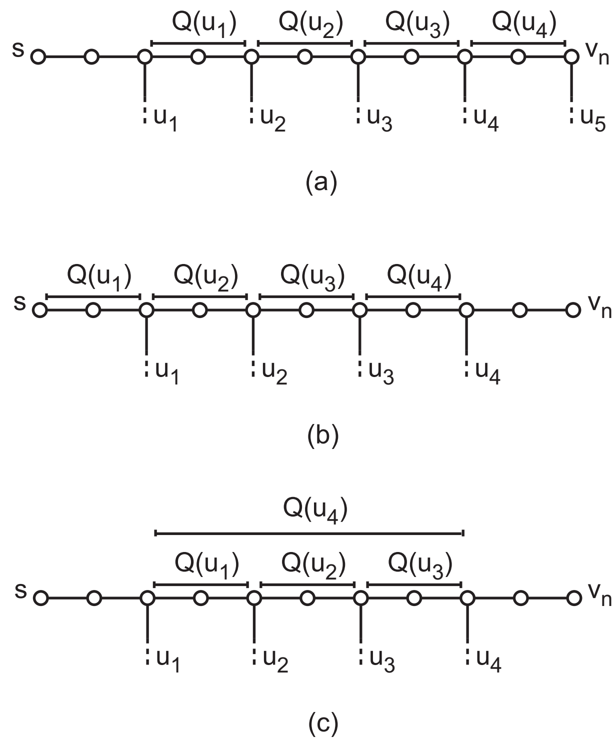

Figure 3.9: The paths $Q(u)$ for a single player i. (a) i does not own s, but has a terminal in U_n (b) i owns s (c) i does not own s or a terminal in U_n

Figure 3.9(a) illustrates what the paths $Q(u)$ for terminals u of i look like if i does not own s and owns a terminal in U_n. In this figure, the terminals u_1 through u_5 are all the terminals of i in any of the sets U_r. Figure 3.9(b) shows the same thing in the case that i owns s. Figure 3.9(c) shows this for the case that i owns neither s nor a terminal of U_n.

Suppose we were able to form payments S_i such that for each terminal u of i with $u \notin U_n$ we can include one edge of $Q(u)$ into S_i, but nothing else. Then the conditions in Lemma 3.5.5 (not involving A) would be satisfied. The proof of this is as follows.

Case 1: i does not own s, and has a terminal in U_n. Then every component of $P \backslash S_i$

contains a terminal of i, since there is a terminal of i between every $Q(u)$ for u belonging to i, as well as before the first such $Q(u)$, and after the last one (since the last such $Q(u)$ ends at v_n, and i has a terminal in U_n).

Case 2: i owns s. Similarly to Case 1, every component of $P \backslash S_i$ contains a terminal of i, since there is a terminal of i between and after every $Q(u)$ for u belonging to i by definition of $Q(u)$. There is a terminal of i before the first such $Q(u)$ because s belongs to i.

Case 3: i does not own s, and has no terminals in U_n. Let u' be the last terminal of i. For the edges of S_i associated with $Q(u)$, $u \neq u'$, the argument is the same as the previous cases. We now have an additional edge in $Q(u')$ which is in S_i. This edge can form at most 1 component of $P \backslash S_i$ which does not contain a terminal of i, as desired.

We have now shown that if we associate each edge of S_i with a path $Q(u)$ for terminals u owned by i so that there is at most 1 edge associated with each $Q(u)$, then the conditions of Lemma 3.5.5 are satisfied. Now suppose we form payments S_i such that for each terminal u of i with $u \notin U_n$ we include several edges of $Q(u)$ into S_i. Suppose these edges are $e_1, e_2, \ldots e_\ell \in Q(u)$ with e_1 being closest to s. As long as the components of $P \backslash S_i$ between e_j and e_{j+1} are coupled, then by the above argument the conditions of Lemma 3.5.5 still hold. Our goal now is to associate with each terminal u of i, $u \notin U_n$, a set of edges $e_1, e_2, \ldots e_\ell \in Q(u)$ so that each component of $P \backslash S_i$ between e_j and e_{j+1} is coupled.

Consider a pair of edges e and e' such that they are contained in exactly the same paths $Q(u)$. Let C be the component of $P \backslash \{e, e'\}$ between e and e'. Every path $Q(u)$ that starts in C must also end in C. This means that if a player i has a terminal $u' \in C$, then all of i's terminals must be in C, since by construction of the paths $Q(u)$, for every terminal of i there is a path $Q(u)$ which starts at that terminal and ends at another terminal of i. C is coupled, therefore. This implies that to prove this lemma, it is enough for us to associate with each terminal $u \notin U_n$ a set of edges $e_1, e_2, \ldots e_\ell \in Q(u)$ so that these edges lie in exactly the same paths $Q(u)$, and so that all edges are covered by this assignment.

Definition 3.5.7 *A* <u>*Q-set*</u> L *is a maximal set of edges of* P *such that for every edge* $e \in L$*, the set of paths* $Q(u)$ *that contain* e *is exactly the same, for* $u \in \cup U_r$*.*

We desire to form a matching between Q-sets and terminals $u \notin U_n$, covering the Q-sets, so that if L is matched to u, then $L \subseteq Q(u)$. Then if we set L to be paid for by the player that owns u, the conditions in Lemma 3.5.5 are satisfied.

We also would like to construct a set A as specified in the statement of Lemma 3.5.5. Notice that in the above case argument about components of $P \backslash S_i$, the certificate terminals if only some paths $Q(u)$ have edges in S_i are as follows. In Case 1, they are the terminals u such that S_i has an edge in $Q(u)$, together with a terminal in U_n. In Case 2, they are the similar terminals together with s. In Case 3, they are the similar terminals together (possibly) with the last terminal of i. Suppose that in the above matching, only a single terminal from each U_r is matched to a Q-set, and set A to be s together with the matched terminals. By this case analysis, for each player i, all the certificate terminals are contained in A or U_n, and for the player that owns s, all the certificate terminals are contained in A (not U_n), as desired in Lemma 3.5.5. The only unclear case if for a player i in Case 3. If the last terminal u' of i is matched to a Q-set and is therefore in A, then we are done. Otherwise, it must be that i has no components of $P \backslash S_i$ which are uncoupled and have no terminals of i, since there is only one Q-set paid for by i contained in each $Q(u)$ for u a terminal of i. Therefore, the right-most component of $P \backslash S_i$ is the only one that may be uncoupled and not contain a terminal of i from A, as desired.

We have now shown that to prove the lemma, it is enough to form a matching between Q-sets and terminals $u \notin U_n$, covering the Q-sets, so that if L is matched to u, then $L \subseteq Q(u)$, and so that only one terminal for each U_r is matched to a Q-set. We do this by constructing the following bipartite graph (Y, Z).

Let Y have a node for each Q-set in P, and let Z be the nodes of P. Form an edge between a node $v_r \in Z$ and node $L \in Y$ if there exists some terminal $u \in U_r$ such that $L \subseteq Q(u)$. For $X \subseteq Y$, define $\partial(X)$ to be the set of nodes in Z which X has edges to. According to Hall's Matching Theorem, there exists a matching in this bipartite graph with all nodes of Y incident to an edge of the matching if for each set $X \subseteq Y$, $|\partial(X)| \geq |X|$. Arrange the edges of the Q-sets of X in the order they appear in P. We want to show that between every Q-set of X, there appears a node belonging to $\partial(X)$.

Consider some edge e of X that is not the first one in P. Suppose this edge belongs to Q-set L, and the previous edge e' in X to some Q-set L'. Since these are different Q-sets, there must be some path $Q(u)$, with an endpoint at node v_j between e' and e, and $u \in U_r$ and $v_r \in \partial(X)$ (since $Q(u)$ contains either e or e'). If u belongs to the same player as s, let $Q(u)$ be the left-most such path that contains e'. This implies that $v_j = v_r$, so $v_j \in \partial(X)$. If u belongs to a player i not owning s, let $Q(u)$ be the right-most such path that contains e, if it exists. Once again, this implies that $v_j = v_r$. If such a path does not exist, this implies that the right-most terminal u' of i is between e and e', and the left-most is not. By the definition of $Q(u')$, it must be that $e' \in Q(u')$, so we have a node of $\partial(X)$ between e' and e.

Now let L be the first Q-set of X that appears in P. By our assumption, $U_1 \neq \{s\}$. Therefore, there must be some $Q(u)$ containing L such that u belongs to a different player than s. By construction of $Q(u)$, $u \in U_r$ for some v_r that comes before L. This means that there is a node of $\partial(X)$ before L.

Therefore, $|X| \leq |\partial(X)|$, and so we can assign a terminal $s_r \in U_r$ to each node v_r, $r = 1 \ldots n-1$, such that these terminals are matched to all the Q-sets of P, with the Q-set matched to s_r contained in $Q(s_r)$. By the above argument, this proves the lemma.

■

3.5.4 Extensions and Lower Bounds

We have now shown that in any game, we can find a 3-approximate Nash purchasing the optimal network. We proved this by constructing a payment scheme so that each player pays for at most 3 connection sets. This is in fact a tight bound. In the example shown in Figure 3.10, there must be players that pay for at least 3 connection sets. There are k players, with only two terminals (s_i and t_i) for each player i. Each player must pay for edges not used by anyone else, which is a single connection set. There are $2k - 3$ other edges, and if a player i pays for any 2 of them, they are 2 separate connection sets, since the component between these 2 edges would be uncoupled and would not contain any terminals of i. Therefore, there must be at least one player that is paying for 3 connection sets.

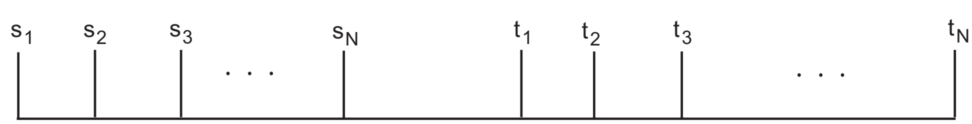

Figure 3.10: A graph where players must pay for at least 3 connection sets.

This example does not address the question of whether we can lower the approximation factor of our Nash equilibrium to something other than 3 by using a method other than connection sets. As a lower bound, in Section 3.6 we give a simple sequence of games such that in the limit, any Nash equilibrium purchasing the optimal network must be at least $(\frac{3}{2})$-approximate.

Polynomial-time algorithm Since the proof of Theorem 3.5.1 is constructive, it actually contains a polynomial-time algorithm for generating a 3-approximate Nash equilibrium on T^*. We can use the ideas from Theorem 3.4.4 to create an algorithm which, given an α-approximate Steiner forest T, finds a $(3+\varepsilon)$-approximate Nash equilibrium which pays for a Steiner forest T' with $c(T') \leq c(T)$, as follows. However, this algorithm requires a polynomial-time optimal Steiner tree finder as a subroutine. We can forego this requirement at the expense of a higher approximation factor.

We start by defining $\gamma = \frac{\varepsilon c(T)}{(1+\varepsilon)n\alpha}$, for ε small enough so that γ is smaller than the minimum edge cost. The algorithm of Theorem 3.5.1 generates at most 3 connection sets for each player i, even if the forest of bought edges is not optimal. We use this algorithm to pay for all but γ of each edge in T. We can check if each connection set is actually cheaper than the cheapest deviation of player i, which is found by the cheapest Steiner tree algorithm. If it is not, we can replace this connection set with the cheapest deviation tree and run this algorithm over again. The fact that we are replacing a connection set means that all the terminals are still connected in the new tree. If we modify T in this manner, it is easy to see that we have decreased its cost by at least γ.

We can now use the arguments from Theorem 3.4.4 to prove that this algorithm produces a $(3+\varepsilon)$-approximate Nash equilibrium, and runs in time polynomial in n, α, and ε^{-1}. It requires a poly-time Steiner tree subroutine, however. If each player only has two terminals, finding the cheapest Steiner tree is the same as finding the cheapest path, so this is possible, and we can indeed find a cheap $(3+\varepsilon)$-approximate Nash equilibrium in polynomial time.

For the case where players may have more than two terminals, we can easily modify the above algorithm to use polynomial time approximations for the optimal Steiner tree, at the expense of a higher approximation factor. If we use a 2-approximate Steiner forest T, and an optimal Steiner tree 1.55-approximation algorithm from [57] as our subroutine, then the above algorithm actually gives a $(4.65+\varepsilon)$-approximate Nash equilibrium on T' with $c(T') \leq 2 \cdot OPT$, in time polynomial in n and ε^{-1}.

3.6 A Lower Bound on the Stability of Optimal Solutions

Claim 3.6.1 *For any $\epsilon > 0$, there is a game such that any equilibrium which purchases the optimal network is at least a $(\frac{3}{2} - \epsilon)$-approximate Nash equilibrium.*

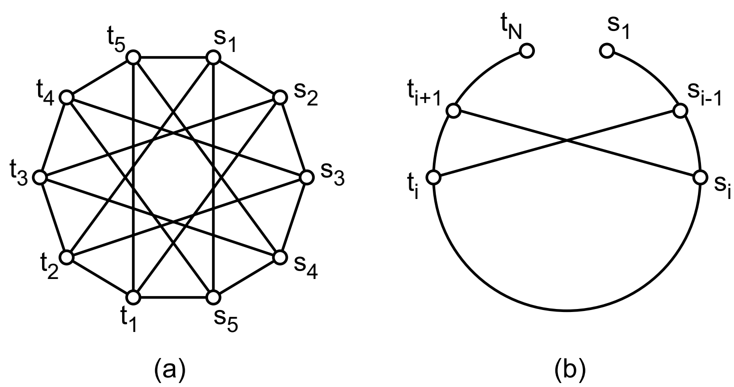

Figure 3.11: A game with best Nash equilibrium on OPT tending to at least a $\frac{3}{2}$-approximation.

Proof : Construct the graph H_k on $2k$ vertices as follows. Begin with a cycle on $2k$ vertices, and number the vertices 1 through $2k$ in a clockwise fashion. For vertex i, add an edge to vertices $i + k - 1 \mod (2k)$ and $i + k + 1 \mod (2k)$. Let all edges have cost 1. Finally, we will add k players with 2 terminals, s_i and t_i, for each player i. At node j, add the label s_j if $j \leq k$ and t_{j-k} otherwise. Figure 3.11(a) shows such a game with $k = 5$.

Consider the optimal network T^* consisting of all edges in the outer cycle except (s_1, t_k). We would like to show that any Nash which purchases this solution must be at least $(\frac{6k-21}{4k-11})$-approximate. This clearly would prove our claim.

First we show that players 1 and k are not willing to contribute too much to any solution that is better than $(\frac{3}{2})$-approximate. Suppose we have such a solution. Define x to be player 1's contribution to his connecting path in T^*, and define y to be his contribution to the remainder of T^*. Thus player 1 has a total payment of $x + y$. Player 1 can deviate to only pay for x. Furthermore, player 1 could deviate to purchase only y and the edge (s_1, t_k). If we have a solution that is at most $(\frac{3}{2})$-approximate, then we have that $\frac{x}{x+y} \geq \frac{2}{3}$ and similarly $\frac{y+1}{x+y} \geq \frac{2}{3}$. Taken together this implies that $\frac{1}{x+y} \geq \frac{1}{3}$, or $x + y \leq 3$. A symmetric argument shows that player k is also unwilling to contribute more than 3.

Thus we have that the remaining $k - 2$ players must together contribute at least $2k - 7$. Therefore there must be some player other than 1 or k who contributes $\frac{2k-7}{k-2}$. Suppose player i is such a player. Let x be the amount that player i contributes to his connecting path in T^*. Let y be his contribution to (s_{i-1}, s_i) and let z be his contribution to (t_i, t_{i+1}). See Figure 3.11(b).

Now consider three possible deviations available to player i. He could choose to contribute only x. He could contribute y and purchase edge (s_{i-1}, t_i) for an additional cost of 1. Or he could contribute z and purchase edge (s_i, t_{i+1}), also for an additional cost of 1. We will only consider these possible deviations, although of course there are others. Note that if i was contributing to any other portion of T^*, then we could remove those contributions and increase x, y, and z, thereby strictly decreasing i's incentive to deviate. Thus we can safely assume that these are i's only payments, and hence

$$x + y + z \geq \frac{2k - 7}{k - 2}.$$

Since i is currently paying at least $x + y + z$, we know that his incentive to deviate is at least

$$\max\left(\frac{x + y + z}{x}, \frac{x + y + z}{y + 1}, \frac{x + y + z}{z + 1}\right).$$

This function is minimized when $x = y + 1 = z + 1$. Solving for x we find that

$$x \geq \frac{4k - 11}{3k - 6}.$$

Thus player i's incentive to deviate is at least

$$\frac{x + y + z}{x} \geq \frac{3x - 2}{x} = 3 - \frac{2}{x} \geq 3 - 2\frac{3k - 6}{4k - 11} = \frac{6k - 21}{4k - 11}.$$

Therefore as k grows, this lower bound on player i's incentive to deviate tends towards $\frac{3}{2}$. Note that in this proof, we only considered one optimal network, namely T^*. If we modify G by increasing the costs of all edges not in T^* by some small $\varepsilon > 0$, then T^* is the only optimal network. Repeating the above analysis under these new costs still yields a lower bound of $\frac{3}{2}$ for the best approximate Nash on T^* in the limit as k grows and ε tends to 0. ■

3.7 NP Completeness

In this section, we present a brief proof that determining the existence of Nash equilibria in a given graph is NP-complete if the number of players is $O(n)$ (where n is the number of nodes in the graph). We present a reduction from 3-SAT to show that the problem is NP-hard. The graph constructed will have unit cost edges.

Consider an arbitrary instance of 3-SAT with clauses C_j and variables x_i. We will have a player for each variable x_i, and two players for each clause C_j. For each variable x_i construct the gadget shown in Figure 3.12a. The source and sink of the player x_i are the vertices s_i and t_i respectively. When player x_i buys the left path or right path, this corresponds to x_i being set to be true or false, respectively. For clarity, we will refer to this player as being the i^{th} variable player.

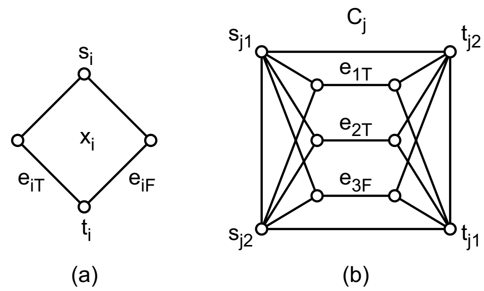

Figure 3.12: Gadgets for the NP-completeness reduction.

Next, we construct a gadget for each clause C_j. The construction is best explained through an example clause $C_j = (x_1 \vee x_2 \vee \overline{x}_3)$ whose gadget is given in Figure 3.12b. The two players for C_j have their source sink pairs as (s_{j1}, t_{j1}) and (s_{j2}, t_{j2}) respectively. We will call both players on this gadget clause players. The final graph is constructed by gluing these gadgets together at the appropriate labeled edges. Specifically, the edges in clause gadget C_j labeled e_{1T}, e_{2T}, and e_{3F} are the same edges that appear in the corresponding variable gadgets. In other words, among all clauses and variable gadgets, there is only one edge labeled e_{iT} and only one labeled e_{iF}, and all the interior nodes in the gadget for each clause C_j are nodes in variable gadgets.

Suppose that there is a satisfying assignment A in our 3-SAT instance. Consider the strategy in which variable player i fully buys the left path if x_i is true in A and fully buys the right path otherwise. Since this is a satisfying assignment, by our construction each clause gadget has at least one interior edge fully paid for by a variable player. For each clause C_j, let e be one such edge, and let both players on this gadget buy the unique path of length 3 that connects their terminals which uses edge e. It is easy to see that the clause players are satisfied as the cost of this path to each clause player is 2, the minimum that he has to pay on any path from source to sink under the current payment scheme. The cheapest deviation for each variable player also costs 2, and therefore they do not have any incentive to move either. Thus, this forms a Nash equilibrium.

Suppose now that there is a Nash equilibrium. We will argue that this Nash equilibrium has to have a specific set of edges paid for. First, note that the contribution of each player is not more than 2, as the length of the shortest path is exactly 2.

Now suppose some perimeter edge of clause C_j is being bought. We know from the example in Figure 3.1 that perimeter edges cannot be bought by the clause players in C_j alone, for that would not constitute a Nash strategy. Therefore there must be some other player, variable or clause, contributing to the perimeter edge of C_j. Also, since this is a Nash strategy, any perimeter edge on which there is a positive contribution by any player must be fully bought. And once any perimeter edge of C_j has a positive contribution from a non-C_j player the payments of both the clause players of C_j will be strictly less than 2 in a Nash strategy.

Suppose one of the clauses, C_j, has some perimeter edge bought. Since at Nash equilibrium, the set of edges bought must form a Steiner forest, we look at the component of the Steiner forest that has the the clause C_j. We will show that the number of edges in this component is more than twice the number of players involved. Then, there must be a player who is paying more than 2, and hence this cannot be a Nash equilibrium.

Suppose there are x clause players and y variable players in the component of the forest containing C_j. We know from the example in Figure 3.1 that $x + y > 2$. Then, the total number of nodes in the Nash component containing C_j is $2x + 3y$ as we have to count the two source-sink nodes for each clause player and the three nodes on the path of each variable player. Since this is a connected tree, the total number of edges in this component is $2x + 3y - 1$. The average payment per player is then given by $\frac{2x+3y-1}{x+y} = 2 + \frac{y-1}{x+y}$. Now if $y > 1$, then the average payment per player is more than 2. Thus there must be some player who is paying more than 2, which is infeasible in a Nash. If $y = 1$, then the average payment per player is exactly 2. But again, since we know that the clause players of C_j pay strictly less than 2 each, there must be some player who pays strictly more than 2, which is again impossible. Lastly, we cannot have $y = 0$ as then whenever a clause player participates in paying for another clause, he must use a node in the path of a variable player, and thereby include this variable player in the component of C_j.

This implies that variable players only select paths within their gadget. Furthermore, it implies that variable players must pay fully for their entire path. Suppose i is a variable player who has selected the left (true) path, but has not paid fully for the second edge in that path. The remainder of this cost must be paid for by some clause player or players. But for such a clause player to use this edge, he must also buy two other edges, which are not used by any other player. Hence such a clause player must pay strictly more than 2. But there is always a path he could use to connect of cost exactly 2, so this can not happen in a Nash equilibrium. Thus we have established that variable players pay fully for their own paths.

Now consider any clause gadget. Since we have a Nash equilibrium, we know that only internal edges are used. But since each clause player can connect his terminals using perimeter edges for a cost of exactly 2, one of the interior variable edges must be bought by a variable player in each clause gadget. If we consider a truth assignment

A in which x_i is true if and only if player i selects the left (true) path, then this obviously satisfies our 3-SAT instance, as every clause has at least one variable forcing it to evaluate to true.

Therefore, this game has a Nash equilibrium if and only if the corresponding formula is satisfiable, and since this problem is clearly in NP, determining whether a Nash equilibrium exists is NP-Complete.

Chapter 4

Network Creation with Fair Cost Sharing

4.1 Introduction

In chapter 3 we introduced the connection game with arbitrary cost sharing. This game can be viewed as being very "unregulated," in the sense that no restrictions are placed on the manner in which users divide up edge costs. As a result, many instances of this game have no pure Nash equilibrium; furthermore, it is often the case that, when pure Nash equilibria do exist, certain users are able to act as "free riders," paying very little or nothing at all. This can lead to situations in which even the best equilibria are far more expensive than centrally designed solutions. In this chapter, we continue our study of competitive network formation, and examine the extent to which we can rectify these problem via mechanism design. More precisely, we will show how a number of properties of the connection game improve when players' strategies are restricted by means of a cost-sharing mechanism.

A cost-sharing mechanism can be viewed as the underlying protocol that determines how much a network serving several participants will cost to each of them. Specifically, suppose that each user i has a set of terminals T_i that it wishes to connect; it chooses a tree S_i connecting the nodes in T_i; and the cost-sharing mechanism then charges user i a cost of $C_i(S_1, \ldots, S_k)$. (Note that this cost can depend on the choices of the other users as well.) Although there are in principle many possible cost-sharing mechanisms, research in this area has converged on a few mechanisms with good theoretical and empirical behavior; here we focus on the following particularly natural one: the cost of each edge is shared equally by the set of all users whose trees contain it, so that

$C_i(S_1, S_2, \ldots, S_k) = \sum_{e \in S_i} \frac{c_e}{|\{j : e \in S_j\}|}$. This equal-division mechanism has a number of basic economic motivations; it can be derived from the Shapley value [53], and it can be shown to be the unique cost-sharing scheme satisfying a number of different sets of axioms [26, 36, 53]. We will refer to it as the *Shapley* or *fair cost-sharing mechanism*. Note that the total edge cost of the designed network is equal to the sum of the costs in the union of all S_i, and the costs allocated to users in the Shapley mechanism completely pay for this total edge cost: $\sum_{i=1}^{n} C_i(S_1, S_2, \ldots, S_k) = \sum_{e \in \cup_i S_i} c_e$.

Now, the general question is to determine how this basic cost-sharing mechanism serves to influence the strategic behavior of the users, and what effect this has on the structure and overall cost of the network one obtains. Given a solution to the network design problem consisting of a vector of trees $(S_1, \ldots, S_k)$ for the n users, user i would be interested in deviating from this solution if there were an alternate tree S_i' that also connected all terminals in T_i such that changing to S_i' would lower its cost under the resulting allocation: $C_i(S_1, \ldots, S_{i-1}, S_i', S_{i+1}, \ldots, S_k) < C_i(S_1, \ldots, S_{i-1}, S_i, S_{i+1}, \ldots, S_k)$. We say that a strategy (a tree S_i for each player i) is a *Nash equilibrium* if no user has an interest in deviating. As we will see below, there exists a set of strategies in Nash equilibrium for every instance of this network design game. (In this paper, we will only be concerned with *pure* Nash equilibrium; i.e. with equilibria where each user deterministically chooses a single tree.)

The goal of a network design protocol is to suggest for each user i a strategy S_i so that the resulting strategy is in Nash equilibrium and its total cost exceeds that of an optimal set of trees by as small a factor as possible; this factor is the *price of stability* of the instance. It is useful at this point to consider a simple example that illustrates how the price of stability can grow to a super-constant value (with k). Suppose k players wish to connect the common source s to a private terminal t_i, assume player i has its

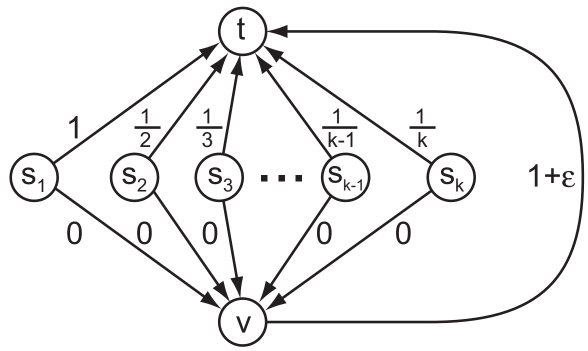

Figure 4.1: An instance in which the price of stability converges to $H(k) = \Theta(\log k)$ as $\varepsilon \to 0$.

own path of cost $1/i$, and all players can share a common path of cost $1 + \varepsilon$ for some small $\varepsilon > 0$ (see Figure 4.1). The optimal solution would connect all agents through the common path for a total cost of $1 + \varepsilon$. However, if this solution were offered to the users, they would defect from it one by one to their alternate paths. The unique Nash equilibrium has a cost of $\sum_{i=1}^{k} \frac{1}{i} = H(k)$.

While the price of stability in this instance grows with k, it only does so logarithmically. It is thus natural to ask how bad the price of stability can be for this network design problem. If we think about the example in Figure 4.1 further, it is also interesting to note that a good Nash equilibrium is reached by iterated greedy updating of players' solutions (in other words, best-response dynamics) starting from an optimal solution; it is natural to ask to what extent this holds in general.

4.1.1 Results

Our first main result is that in every instance of the network design problem with Shapley cost-sharing, there always exists a Nash equilibrium of total cost at most $H(k)$ times optimal. In other words, the simple example in Figure 4.1 is in fact the worst possible case.

We prove this result using a *potential function* method due to Monderer and Shapley [52] and Rosenthal [59] (see also [11]): one defines a potential function Φ on possible solutions and shows that any improving move by one of the users (i.e. to lower its own cost) reduces the value of Φ. Since the set of possible solutions is finite, it follows that any sequence of improving moves leads to a Nash equilibrium. The goal of Monderer and Shapley's and Rosenthal's work was to prove existence statements of this sort; for our purposes, we make further use of the potential function to prove a bound on the price of stability. Specifically, we give bounds relating the value of the potential for a given solution to the overall cost of that solution; if we then iterate best-response dynamics starting from an optimal solution, the potential does not increase, and hence we can bound the cost of any solution that we reach. Thus, for this network design game, best-response dynamics starting from the optimum does in fact always lead to a good Nash equilibrium.

We can extend our basic result to a number of more general settings. To begin with, the $H(k)$ bound on the price of stability extends directly to the case in which users are selecting arbitrary subsets of a ground set (with elements' costs shared according to the Shapley value), rather than sub-trees in a graph; it also extends to the case in which the cost of each edge is a non-decreasing concave function of the number of users on it. In addition, our results also hold if we introduce capacities into our model; each edge e may be used by at most u_e players, where u_e is the capacity of e.

We arrive at a more technically involved set of extensions if we wish to add latencies to the network design problem. Here each edge has a concave *construction cost* $c_e(x)$ when there are x users on the edge, and a *latency cost* $d_e(x)$; the cost experienced by a user is the full latency plus a fair share of the construction cost, $d_e(x)+c_e(x)/x$. We give general conditions on the latency functions that allow us to bound the price of stability in this case at $d \cdot H(k)$, where d depends on the delay functions used. Moreover, we obtain stronger bounds in the case where users experience only delays, not construction costs; this includes a result that relates the cost at the best Nash equilibrium to that of an optimum with twice as many players, and a result that improves the potential-based bound on the price of stability for the single-source delay-only case.

Since a number of our proofs are obtained by following the results of best-response dynamics via a potential function, it is natural to investigate the speed of convergence of best-response dynamics for this game. We show that it converges to a Nash equilibrium in polynomial time for the case of two players, but that with k players, it can run for a time exponential in k. Whether there is a way to schedule players' moves to make best-response converge in a polynomial number of steps for this game in general is an interesting open question.

Finally, we consider a natural generalization of the cost-sharing model that carries us beyond the potential-function framework. Specifically, suppose each user has a *weight* (perhaps corresponding to the amount of traffic it plans to send), and we change the cost-allocation so that user i's payment for edge e is equal to the ratio of its weight to the total weight of all users on e. In addition to being intuitively natural, this definition is analogous to certain natural generalizations of the Shapley value [51]. The weighted model, however, is significantly more complicated: there is no longer a potential function whose value tracks improvements in users' costs when they greedily update their solutions. We show that for weighted games, best-response dynamics converge in the case that all users seek to construct a path from a node s to a node t (the price of stability here is 1), and in the general model of users selecting sets from a ground set, when each element appears in the sets of at most two users. We also present a construction involving user weights that grow exponentially in k that demonstrates that the price of stability in weighted games can be as high as $\Omega(k)$.

4.1.2 Related Work

The bulk of the work on cost-sharing (see e.g. [26, 36] and the references there) tends to assume a fixed underlying set of edges. Jain and Vazirani [37] and Kent and Skorin-Kapov [43] consider cost-sharing for a single source network design game. Cost-sharing games assume that there is a central authority that designs and maintains the network, and decides appropriate cost-shares for each agent, depending on the graph and all other agents, via a complex algorithm. The agents' only role is to report their utility for being included in the network.

Here, on the other hand, we consider a simple cost-sharing mechanism, the Shapley-value, and ask what the strategic implications of a given cost-sharing mechanism are for the way in which a network will be designed. This question explores the feedback between the protocol that governs network construction and the behavior of self-interested agents that interact with this protocol. An approach of a similar style, though in a different setting related to routing, was pursued by Johari and Tsitsiklis [40]; there, they assumed a network protocol that priced traffic according to a scheme due to Kelly [42], and asked how this protocol would affect the strategic decisions of self-interested agents routing connections in the network.

The special case of our game with only delays is closely related to the congestion game of [63, 61]. They consider a game where the amount of flow carried by an individual user is infinitesimally small (a *non-atomic game*), while in this paper we assume that each user has a unit of flow, which it needs to route on a single path. In the non-atomic game of [63, 61] the Nash equilibrium is essentially unique (hence there is no distinction between the price of anarchy and stability), while in our atomic game there can be many equilibria. Fabrikant, Papadimitriou, and Talwar [24] consider our atomic game with delays only. They give a polynomial time algorithm to minimize the potential function Φ in the case that all users share a common source, and show that finding any equilibrium solution is PLS-complete for multiple source-sink pairs. Our results extend the price of anarchy results of [63, 61] about non-atomic games to results on the price of stability for the case of single source atomic games.

A weighted game similar to our is presented by Libman and Orda [46], with a different mechanism for distributing costs among users. They do not consider the price of stability, and instead focus on convergence in parallel networks. Subsequent work on our weighted game by Chen and Roughgarden [14] has shown that in fact, pure Nash equilibria do not always exist. They use a relaxation of a potential function to prove that weighted games do admit $\log(w_{max})$-approximate equilibria, where the price of stability is logarithmic in the total weight of all players, and w_{max} is the largest weight of any player. They further prove that this result is nearly the best possible.

4.2 Nash Equilibria of Network Design with Shapley Cost-Sharing

We now formally define the *Connection Game with Fair Cost-Sharing*, which we will also refer to simply as the *Fair Connection Game*. We let k denote the number of players and consider a directed graph $G = (V, E)$, where each edge has a nonnegative cost c_e. Each player i has a set of terminal nodes T_i that he wants to connect. A strategy of a player is a set of edges $S_i \subset E$ such that S_i connects all nodes in T_i. We assume that we use the Shapley value to share the cost of the edges, i.e. all players using an edge split up the cost of the edge equally. Given a vector of players' strategies $S = (S_1, \ldots, S_k)$, let x_e be the number of agents whose strategy contains edge e. Then the cost to agent

i is $C_i(S) = \sum_{e \in S_i}(c_e/x_e)$, and the goal of each agent is to connect its terminals with minimum total cost.

In the worst case, Nash equilibria can be very expensive in this game, so that the price of anarchy becomes as large as k. To see this, consider k players with common source s and sink t, and two parallel edges of cost 1 and k. The worst equilibrium has all players selecting the more expensive edge, thereby paying k times the cost of the optimal network. However, we can bound the price of stability by $H(k)$, which is the harmonic sum $1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{k}$, as follows.

Theorem 4.2.1 *The price of stability of the fair connection game is at most $H(k)$.*

Proof : The fair connection game that we have defined falls into the class of congestion games as defined by Monderer and Shapley [52], as the cost of an edge e to a user i is $f_e(x) = c_e/x$, which depends only on edge e and the number of users x whose strategy contains e. Monderer and Shapley [52] show that all congestion games have deterministic Nash equilibria. They prove this using a potential function Φ, defined as follows.

$$\Phi(S) = \sum_{e \in E} \sum_{x=1}^{x_e} f_e(x) \tag{4.1}$$

Monderer and Shapley [52] show that for any strategy $S = (S_1, \ldots, S_k)$ if a single player i deviates to strategy S_i', then the change in the potential value $\Phi(S) - \Phi(S')$ of the new strategy set $S' = (S_1, \ldots, S_i', \ldots, S_k)$ is exactly the change in the cost to player i. Note that the change of player i's strategy affects the cost of many other players $j \neq i$, but the Φ value is not effected by the change in the cost of these players, it simply tracks the cost of the player who changes its strategy. They call a game in which such a function Φ exists a *potential game.* To show that such a potential game has a deterministic Nash equilibrium, start from any state $S = (S_1, \ldots, S_k)$ and consider a sequence of selfish moves (allowing players to change strategies to improve their costs). In a congestion game any sequence of such improving moves leads to a Nash equilibrium as each move decreases the potential function Φ, and hence must lead to a stable state.

Monderer and Shapley do not say anything about the quality of Nash equilibria with respect to the centralized optimum, but we can use their potential function to establish our bound. Let x_e be defined as above with respect to S. Now the potential function of Equation 4.1 in our case is $\Phi(S) = \sum_{e \in E} c_e H(x_e)$. According to the above argument, any improving deviation decreases $\Phi(S)$, and so a sequence of improving deviations by players must eventually result in a Nash equilibrium.

Consider the strategy $S^* = (S_1^*, \ldots, S_k^*)$ defining the optimal centralized solution. Let $OPT = \sum_{e \in S^*} c_e$ be the cost of this solution. Then, $\Phi(S^*) \leq \sum_{e \in S^*}(c_e \cdot H(k))$, which is exactly $H(k) \cdot OPT$. Now we start from strategy S^* and follow a sequence of improving self-interested moves. We know that this will result in a Nash equilibrium S with $\Phi(S) \leq \Phi(S^*)$.

Note that the potential value of any solution S is at least the total cost: $\Phi(S) \geq \sum_{e \in S} c_e = cost(S)$. Therefore, there exists a Nash equilibrium with cost at most $H(k) \cdot OPT$, as desired. ■

Recall from the Introduction that this bound is tight as shown by the example in Figure 4.1. Unfortunately, even though Theorem 4.2.1 says that cheap Nash equilibria exist, finding them is NP-complete.

Theorem 4.2.2 *Given an instance of a fair connection game and a value C, it is NP-hard to determine if the game has a Nash equilibrium of cost at most C.*

Proof : The reduction is from 3D-Matching. Given an instance of 3D-Matching with node sets X, Y, Z, form a graph $G = (V, E)$ as follows. Form a node for each node in

X, Y, and Z, and also a node $v_{i,j,k}$ for each 3D edge (x_i, y_j, z_k). Also add an additional node t. Form a directed edge from each node $v_{i,j,k}$ to t with cost function $c_e = 3$. Form a directed edge from each node v in X, Y, Z to all nodes representing 3D edges that contain v. Make these edges have a cost $c_e = 0$. Let $k = |X| + |Y| + |Z|$, and form a player for each node in v in $X \cup Y \cup Z$. This player has two terminals: v and t.

If there exists a 3D Matching in the 3D-Matching instance, then there exists a Nash equilibrium in the above fair connection game of cost k: Take the 3D Matching M, and let S_i for the player whose terminals are v and t be the edge from v to the unique node $v_{i,j,k}$ corresponding to the 3D edge in M, and the edge from this node to t. Since M is a matching, the cost of S is exactly $3k/3 = k$. S is a Nash equilibrium, since any deviation for a player involves paying for some edge of cost 3 by himself, while the current amount he is paying is 1.

If no 3D Matching exists, then any solution to the fair connection game must cost more than k. Therefore, no Nash equilibrium can exist of cost at most k. This finishes the proof. ■

Notice that the same proof works to show that determining if there exists a Nash equilibrium that costs OPT is NP-complete.

We can extend the results of Theorem 4.2.1 to concave cost functions. Consider the extended fair connection game where instead of a constant cost c_e, each edge has a cost which depends on the number of players using that edge, $c_e(x)$. We assume that $c_e(x)$ is a nondecreasing, concave function, modeling the buy-at-bulk economy of scale of buying edges that can be used by more players. Notice that the cost of an edge $c_e(x)$ might increase with the number of players using it, but the cost per player $f_e(x) = c_e(x)/x$ decreases if $c_e(x)$ is concave.

Theorem 4.2.3 *Take a fair connection game with each edge having a nondecreasing* *concave cost function $c_e(x)$, where x is the number of players using edge e. Then the price of stability is at most $H(k)$.*

Proof : The proof is analogous to the proof of Theorem 4.2.1. We use the potential function $\Phi(S)$ defined by (4.1). As before, the change in potential if a player i deviates equals exactly to the change of that player's payments. We start with the strategy S^* with minimum total cost, and perform a series of improving deviations until we reach a Nash equilibrium S with $\Phi(S) \leq \Phi(S^*)$. To finish the proof all we need to show is that $cost(S) \leq \Phi(S) \leq H(k) \cdot cost(S)$ for all strategies S. The second inequality follows since $c_e(x)$ is nondecreasing and therefore $\sum_{x=1}^{x_e}(c_e(x)/x) \leq H(x_e) \cdot c_e(x_e)$. To see that $cost(S) \leq \Phi(S)$ notice that since $c_e(x)$ is concave, the cost per player must decrease with x, i.e. $c_e(x)/x$ is a nonincreasing function. Therefore, $cost(S) = \sum_{e \in S} c_e(x_e) = \sum_{e \in S} x_e \cdot (c_e(x_e)/x_e) \leq \Phi(S)$, which finishes the proof. ■

Notice that the condition that cost functions be concave is general enough to encompass the utility function of a player being a combination of the cost he has to pay for his edges and the distance between his terminals in the network of bought edges. If c_e is the cost function of an edge, we simply set $c'_e(x) = c_e(x) + x$. The payment of each player i now becomes $|S_i| + \sum_{e \in S_i}(c_e(x_e)/x_e)$, and c'_e is still concave if c_e is concave.

Extensions The proof of Theorem 4.2.3 extends to a general congestion game, where players attempt to share a set of resources R they need. Instead of having an underlying graph structure, we now think of each $s \in R$ as a resource with a concave cost function $c_s(x)$ of the number of users selecting sets containing s. The possible strategies of each player i is a set $\mathcal{S}_i$ of subsets of R. Each player seeks to select a set $S_i \in \mathcal{S}_i$ so as to minimize his cost. Since the proofs above did not rely on the graph structure, they translate directly to this extension.

We can further extend the results to the case when the cost to a player is a combination of the cost $c_e(x)/x$, and a function of the selected set, such as the distance between terminals in the network design case. More precisely, the price of stability is still at most $H(k)$ if each player is trying to minimize the cost $\sum_{e \in S_i}(c_e(x_e)/x_e) + d_i(S_i)$ where c_e is monotone increasing and concave, and d_i is an arbitrary function specific to player i (e.g. a distance function, or diameter of S_i, etc.). The proof is analogous to Theorem 4.2.3, except with a new potential $\Phi(S) = \sum_i d_i(S_i) + \sum_{e \in S} \sum_{x=1}^{x=x_e} \frac{c_e(x)}{x}$. Notice that this is technically not a congestion game on the given graph G. Finally we note that all these results (as well as those subsequent) hold in the presence of capacities. Adding capacities u_e to each edge e and disallowing more than u_e players to use e at any time does not substantially alter any of our proofs.

The Case of Undirected Graphs. While the bound of $H(k)$ is tight for general directed graphs, it is not tight for undirected graphs. Finding the correct bound is an interesting open problem. In the case of two players, our bound on the price of stability is $H(2) = 3/2$. In Section 4.4 we show that that this bound can be improved to $4/3$ in the case of two players and a single source. We also give an example to show that this bound is tight.

4.3 Dealing with Delays

In most of the previous section, we assumed that the utility of a player depends only on the cost of the edges he uses. What changes if we introduce latency into the picture? We have extended this to the case when the players' cost is a combination of "design" cost and the length of the path selected. More generally, delay on an edge does not have to be simply the "hop-count", but can also depend on congestion, i.e., on the number of players using the edge. In this section we will consider such a model.

Assume that each edge has both a cost function $c_e(x)$ and a latency function $d_e(x)$, where $c_e(x)$ is the cost of building the edge e for x users and the users will share this cost equally, while $d_e(x)$ is the delay suffered by users on edge e if x users are sharing the edge. The goal of each user will be to minimize the sum of his cost and his latency. If we assume that both the cost and latency for each edge depend only on the number of players using that edge, then this fits directly into our model of a congestion game above: the total cost felt by each user on the edge is $f_e(x) = c_e(x)/x + d_e(x)$. If the function $xf_e(x)$ is concave then Theorem 4.2.3 applies. But while concave functions are natural for modeling cost, latency tends to be convex.

4.3.1 Combining Costs and Delays

First, we extend the argument in the proof of Theorem 4.2.3 to general functions f_e. The most general version of this argument is expressed in the following theorem.

Theorem 4.3.1 *Consider a fair connection game with arbitrary edge-cost functions f_e. Suppose that $\Phi(S)$ is as in Equation 4.1, with $cost(S) \leq A \cdot \Phi(S)$, and $\Phi(S) \leq B \cdot cost(S)$ for all S. Then, the price of stability is at most $A \cdot B$.*

Proof : Let S^* be a strategy such that S_i^* is the set of edges i uses in the centralized optimal solution. We know from above that if we perform a series of improving deviations on it, we must converge to a Nash equilibrium S' with potential value at most $\Phi(S^*)$. By our assumptions, $cost(S') \leq A \cdot \Phi(S') \leq A \cdot \Phi(S^*) \leq AB \cdot cost(S^*) = AB \cdot OPT$.
■

Our main interest in this section are functions $f_e(x)$ that are the sums of the fair share of a cost and a delay, i.e., $f_e(x) = c_e(x)/x + d_e(x)$. We will assume that $d_e(x)$ is monotone increasing, while $c_e(x)$ is monotone increasing and concave.

Corollary 4.3.2 *If $c_e(x)$ is concave and nondecreasing, $d_e(x)$ is nondecreasing for all e, and $x_e d_e(x_e) \leq A \sum_{x=1}^{x_e} d_e(x)$ for all e and x_e, then the price of stability is at most $A \cdot H(k)$. In particular, if $d_e(x)$ is a polynomial with degree at most l and nonnegative coefficients, then the price of stability is at most $(l+1) \cdot H(k)$.*

Proof : For functions $f_e(x) = c_e(x)/x + d_e(x)$, both the cost and potential of a solution come in two parts corresponding to the cost c and delay d.

For the part corresponding to cost the potential over-estimates the cost by at most a factor of $H(k)$ as proved in Theorem 4.2.3. If on the delay, the potential underestimates the cost by at most a factor of A, then we get the bound of $A \cdot H(k)$ for the price of stability by Theorem 4.3.1. ■

Therefore, for reasonable delay functions, the price of stability cannot be too large. In particular, if the utility function of each player depends on a concave cost and delay that is independent of the number of users on the edge, then we get that the price of stability is at most $H(k)$ as we have shown at the end of the previous section. If the delay grows linearly with the number of users, then the price of stability is at most $2H(k)$.

4.3.2 Games with Only Delays

In this subsection we consider games with only delay. We assume that the cost of a player for using an edge e used by x players is $f_e(x) = d_e(x)$, and d_e is a monotone increasing function of x. This cost function models delays that are increasing with congestion.

We will mostly consider the special case when there is a common source s. Each player i has one additional terminal t_i, and the player wants to connect s to t_i via a directed path. Fabrikant, Papadimitriou, and Talwar [24] showed that in this case, one can compute the Nash equilibrium minimizing the potential function Φ via a minimum cost flow computation. For each edge e they introduce many parallel copies, each with capacity 1, and cost $d_e(x)$ for integers $x > 0$. We will use properties of a minimum cost flow for establishing our results.

A Bicriteria Result

First we show a bicriteria bound, and compare the cost of the cheapest Nash equilibrium to that of the optimum design with twice as many players.

Theorem 4.3.3 *Consider the single source case of a congestion game with only delays. Let S be the minimum cost Nash equilibrium and S^* be the minimum cost solution for the problem where each player i is replaced by two players. Then $cost(S) \leq cost(S^*)$.*

Proof : Consider the Nash equilibrium obtained by Fabrikant et al [24] via a minimum cost flow computation. Assume that x_e is the number of users using edge e at this equilibrium. By assumption, all users share a common source s. Let $D(v)$ denote the cost of the minimum cost path in the residual graph from s to v. The length of the path of user i is at most $D(t_i)$ (as otherwise the residual graph would have a negative cycle) and hence we get that $cost(S) \leq \sum_i D(t_i)$.

Now consider a modified delay function $\hat{d}_e$ for each edge $e = (u, v)$. Define $\hat{d}_e(x) = d_e(x)$ if $x > x_e$, and $\hat{d}_e(x) = D(v) - D(u)$ if $x \leq x_e$. Note that for any edge e we have $D(v) - D(u) \leq d_e(x_e + 1)$ as the edge $e = (u, v)$ is in the residual graph with cost $d_e(x_e + 1)$. This implies that the modified delay $\hat{d}$ is monotone. For edges with $x_e \neq 0$

we also have that $d_e(x_e) \le D(v) - D(u)$ as the reverse edge (v, u) is in the residual graph with cost $-d_e(x_e)$, so the delay of an edge is not decreased.

Now observe that, subject to the new delay $\hat{d}$, the shortest path from s to t_i is length $D(t_i)$ even in an empty network. The minimum possible cost of two paths from s to t_i for the two users corresponding to user i is then at least $2D(t_i)$ for each player i. Therefore the minimum cost of a solution with delays $\hat{d}$ is at least $2\sum_i D(t_i)$.

To bound $cost(S^*)$ we need to bound the difference in cost of a solution when measured with delays $\hat{d}$ and d. Note that for any edge $e = (u, v)$ and any number x we have that $x\hat{d}_e(x) - xd_e(x) \le x_e(D(v) - D(u))$, and hence the difference in total cost is at most $\sum_{e=(u,v)} x_e(D(v) - D(u)) = \sum_i D(t_i)$. Using this, we get that $cost(S^*) \ge \sum_i D(t_i) \ge cost(S)$. ■

Note that a similar bound is not possible for a model with both costs and delays, when additional users compensate to some extent for the price of stability. Consider a problem with two parallel links e and e' and k users. Assume on link e the cost is all design cost $c_e(x) = 1 + \varepsilon$ for a small $\varepsilon > 0$. On the other link e' the cost is all delay, and the delay with x users is $d_{e'}(x) = 1/(k - x + 1)$. The optimum solution is to use the first edge e, and it costs $1 + \varepsilon$. Note that the optimum with any number of extra users costs the same, as this is all design cost. On the other hand, the only Nash is to have all users on e', incurring delay 1, for a total cost of k.

Bounding the Price of Stability with Only Delays

Note that the $H(k)$ term in Corollary 4.3.2 comes from the concave cost c, and so the bound obtained there improves by an $H(k)$ factor when the cost consists of only delay. The results from Corollary 4.3.2 already tell us that if the delay functions are such that $x_e d_e(x_e) \le A\sum_{x=1}^{x_e} d_e(x)$, the the price of stability is at most A. Specifically, we know that if the delays are polynomial of degree l, then the price of stability is at most $l + 1$, and therefore with linear delays the price of stability is at most 2.

Roughgarden [61] showed a tighter bound for non-atomic games. He assumed that the delay is monotone increasing, and the total cost of an edge $xd_e(x)$ is a convex function of the traffic x. He showed that for any class of such functions $\mathcal{D}$ containing all constant functions, the price of anarchy is always obtained on a two node, two link network. Let us call $\alpha(\mathcal{D})$ the price of anarchy for non-atomic games with delays from the class $\mathcal{D}$ (which is also the price of stability, since the Nash equilibrium is unique in that context). For example, Roughgarden [61] showed that for polynomials of degree at most l this bound is $O(l/\log l)$, and for linear delays it is 4/3. Here we extend this result to a single source atomic game, and thereby show tighter bounds than in Corollary 4.3.2 for the singe source case.

Theorem 4.3.4 *If in a single source fair connection game all costs are delays, and all delays are from a set $\mathcal{D}$ satisfying the above condition, then the price of stability is at most $\alpha(\mathcal{D})$.*

Proof : As in the proof of Theorem 4.3.3 consider the Nash equilibrium obtained via a minimum cost flow computation, and let $D(v)$ be the length of the shortest path from s to v in the residual graph. As before we have that $cost(S) \leq \sum_i D(t_i)$. Further, for each edge $e = (u, v)$ we have that $D(v) - D(u) \leq d_e(x_e + 1)$, and for edges with $x_e \neq 0$, we also have that $d_e(x_e) \leq D(v) - D(u)$.

To prove the bound, we consider the modified network $\hat{G}$. Note that the new capacity and the added links do not effect the equilibrium, as $d_e(x_e) \leq D(v) - D(u)$. For each edge e, the two parallel copies: edge e with new capacity x_e and edge e', can carry any number of paths at least as cheaply as the original edge e could since $D(v) - D(u) \leq$

$d_e(x_e+1)$, hence this change in the network can only improve the minimum possible cost. We will prove the bound in this new network by comparing the cost of the Nash equilibrium with the minimum possible cost of a (possibly fractional) flow carrying one unit of flow from s to each of the terminals t_i.

The nice property of $\hat{G}$ is that the optimum fractional flow $\hat{x}$ in $\hat{G}$ is easy to determine. Consider an edge $e=(u,v)$ that is used by $x_e \neq 0$ paths in the equilibrium. We will obtain a fractional flow $\hat{x}_e$ by splitting the corresponding x_e amount of flow between the two edges e and e'. For an edge e let $\ell_e(x)=d_e(x)+xd'_e(x)$. By assumption, $d_e(x) \leq \ell_e(x)$ for all x. For an edge e such that $\ell_e(x_e) \leq D(v)-D(u)$, we set $\hat{x}_e = x_e$, and $\hat{x}_{e'}=0$. Otherwise, let $\hat{x}_e$ be such that $\ell_e(\hat{x}_e)=D(v)-D(u)$, and let $\hat{x}_{e'}=x_e-\hat{x}_e$.

First, we claim that $\hat{x}$ is the minimum cost fractional solution in $\hat{G}$. For all edges $e=(u,v)$ such that $\hat{x}_e \neq x_e$, we have that $\ell_e(\hat{x}_e)=D(v)-D(u)$. When $\hat{x}_e = x_e$, then we have that flow $\hat{x}_e$ is equal the capacity of the edge, and $\ell_e(\hat{x}_e) \leq D(v)-D(u)$. Therefore, if there is a negative cycle in the residual graph of $\hat{x}_e$ with constant edge costs $\ell_e(x_e)$ for e and costs $D(v)-D(u)$ for e', then this is also a negative cost cycle in G with constant edge costs $D(v)-D(u)$. This contradicts x_e being a min-cost flow with those costs, however. We can now use Lemma 4.3.5 to see that $\hat{x}_e$ is also a min-cost flow for edge costs $xd_e(x)$.

The theorem then follows, as on each original edge $e \in E$ the cost $x_e d_e(x_e)$ is at most $\alpha(\mathcal{D})$ times the cost of the corresponding to edges e and e' in $\hat{G}$ by Lemma 4.3.6.

■

To finish the proof of the Theorem, we require the following lemmas.

Lemma 4.3.5 *Let G be a network, and x_e be a fractional flow sending one unit of flow from the source s to each sink t_i. Let ℓ denote the gradient of the total cost $xd_e(x)$, that is, let $\ell_e(x)=d_e(x)+xd'_e(x)$ for each edge e. The flow x_e is minimum cost subject to* *the cost* $\sum_e xd_e(x)$ *if and only if it is a minimum cost flow subject to the constant cost function* $c_e = \ell_e(x_e)$.

Proof : If the flow x_e is not of minimum cost subject to costs c_e, then the residual graph has a negative cycle, and moving a small amount of flow along the cycle decreases the cost $\sum_e xd_e(x)$, as the cost c_e is exactly the gradient of this objective function. To see the other direction, we use the fact that the cost function is convex by assumption, and hence all local optima are also global optima. ■

As with the proof in [61] we start by considering two parallel links; an edge e, which has delay $d(x)$, that will carry some r units of flow, and a parallel link e' with constant delay $d'(x) = d(r)$, independent of the traffic on e'. Now the unique Nash equilibrium is to route all r units of flow on e, while we get the optimum by setting x such that the gradient $c(x) = d(x) + xd'(x)$ is equal to $d(r)$, and sending x units of flow along e, and the remainder $r - x$ along edge e'. Thus we have the following lemma.

Lemma 4.3.6 *If a set* $\mathcal{D}$ *of delay functions satisfies the above condition, then the price of stability is at most* $\alpha(D) = \max_{r,x,d\in\mathcal{D}} rd(r)/(xd(x)+(r-x)d(r))$*, and the maximum is achieved by setting* x *such that* $d(x) + xd'(x) = d(r)$.

4.4 The Undirected Case

While the bound of $H(k)$ on the price of stability is tight for general directed graphs with costs, it is not tight for undirected graphs. Finding the correct bound is an interesting open problem. In the case of two players, our bound on the price of stability is $H(2) = 3/2$. We now show that that this bound can be improved to $4/3$ in the case of two players and a single source.

Here is an example of an undirected two-player game with the price of stability approaching $4/3$. Let G have 3 nodes: s, t_1, and t_2. Player 1 wants to connect t_1 with s, and player 2 wants to connect t_2 with s. There are edges (s, t_1) and (s, t_2) with cost 2. There is an edge (t_1, t_2) with cost $1 + \varepsilon$. The optimal centralized solution has cost $3 + \varepsilon$. However, the cheapest Nash has cost 4. This example implies that the following claim is tight.

Claim 4.4.1 *The price of stability is at most $4/3$ in a fair connection game with two players in an undirected graph, each having two terminals with one terminal in common.*

Proof : Let s be the common terminals, and let t_1 and t_2 be the personal terminals. Consider the optimal centralized solution (S_1, S_2). Let $X_1 = S_1 \backslash S_2$ be the edges only being used by player 1, $X_2 = S_2 \backslash S_1$ be the edge only used by player 2, and $X_3 = S_1 \cap S_2$ be the edges shared by the two players. Let (S_1', S_2') be a Nash equilibrium that a series of improving responses converges to starting with (S_1, S_2). Similarly, let $Y_1 = S_1' \backslash S_2'$, $Y_2 = S_2' \backslash S_1'$, and $Y_3 = S_1' \cap S_2'$. Finally, set $x_i = cost(X_i)$ and $y_i = cost(Y_i)$ for $1 \leq i \leq 3$. By the properties of $\Phi(S_1, S_2)$ from above (more description), we know that $\Phi(S_1', S_2') \leq \Phi(S_1, S_2)$. Substituting in the definition of Φ, we obtain that

$$y_1 + y_2 + \frac{3}{2} y_3 < x_1 + x_2 + \frac{3}{2} x_3. \tag{4.2}$$

Look at S_1' and S_2' as paths instead of sets of edges (there will be no cycles since then this would not be a Nash). We now show that in (S_1', S_2'), as in any Nash equilibrium, once the paths of the two players merge, they do not separate again. Suppose to the contrary that this happens. Let v be the first node that S_1' and S_2' have in common, and set P_1 and P_2 be the subpaths of S_1' and S_2 after v, respectively. We know that $cost(P_1 \backslash P_2) = cost(P_2 \backslash P_1)$, since if they were not equal, say $cost(P_1 \backslash P_2) > cost(P_2 \backslash P_1)$, then player 1 could deviate to P_2 instead and pay strictly less. However, even if they are equal, player 1 could deviate to use P_2 instead of P_1, and pay strictly less, since he will pay the same as before on edges in $P_1 \cap P_2$, and pay only $cost(P_1 \backslash P_2)/2$ in total on the other edges. Therefore, the only way this could be a Nash equilibrium is if $P_1 \cap P_2 = P_1 = P_2$, as desired.

Consider a deviation from (S_1', S_2') that player 1 could make. He could decide to use $X_1 \cup X_2 \cup Y_2 \cup Y_3$ instead of $S_1' = Y_1 \cup Y_3$. This is a valid deviation because player 1 still connects his terminals by following X_1 until X_1 meets with X_2, then following X_2 back to t_2, and then following S_2' to s. Since (S_1', S_2') is a Nash equilibrium, this deviation must cost more to player 1 than his current payments, and so $x_1 + x_2 + y_2/2 + y_3/2 \geq y_1 + y_3/2$. By symmetric reasoning, $x_1 + x_2 + y_1/2 + y_3/2 \geq y_2 + y_3/2$. If we add these inequalities together, we obtain that

$$y_1/2 + y_2/2 \leq 2x_1 + 2x_2. \tag{4.3}$$

To show that the price of stability is at most $4/3$, it is enough to show that $cost(S_1', S_2') \leq \frac{4}{3} cost(S_1, S_2)$. Using the above notation, this is the same as showing $3y_1 + 3y_2 + 3y_3 \leq 4x_1 + 4x_2 + 4x_3$. We do this by using Inequalities 4.2 and 4.3 as follows:

$$\begin{aligned} 3y_1 + 3y_2 + 3y_3 &\leq 3y_1 + 3y_2 + 4y_3 \\ &= \frac{1}{3}(y_1 + y_2) + \frac{8}{3}(y_1 + y_2 + \frac{3}{2}y_3) \\ &\leq \frac{4}{3}(x_1 + x_2) + \frac{8}{3}(x_1 + x_2 + \frac{3}{2}x_3) \\ &= 4x_1 + 4x_2 + 4x_3 \end{aligned}$$

■

4.5 Convergence of Best Response

In this section, we address the convergence properties of best response dynamics in our game. Notice that even though we have shown above that best response dynamics always converge to a Nash equilibrium within a factor of $\log$ of the optimal centralized solution, there are cases where best response dynamics actually start out close to the optimum Nash equilibrium and yet end up in a costly one (although still within a $\log$ factor). For example, consider the same graph as in Figure 4.1, but with an extra path that can be shared by all players of cost 1. If the starting configuration is that all players are using the path of cost $1+\varepsilon$, then there is a sequence of best responses which ends up in a Nash equilibrium of cost $\log k$ (where k is the number of players), even though the starting configuration was of very similar cost to both the centralized optimum and the best Nash equilibrium.

Theorem 4.5.1 *In the two player fair connection game, best response dynamics starting from any configuration converges to a Nash equilibrium in polynomial time.*

Proof : Suppose we start from any configuration C_0. Suppose for $i \geq 1$, the configurations $\{C_i\}$ are obtained by alternating the best responses of the two players. $P_i(1,2)$ refers to the shared path of the two players.

We show inductively that for $i \geq 2$, $P_i(1,2)$ is a contiguous path and that $P_{i+1}(1,2) \supseteq P_i(1,2)$. The base case is showing that $P_2(1,2)$ is a contiguous path. Without loss of generality, assume that the sequence of best responses are as follows

$$C_0 \xrightarrow{2} C_1 \xrightarrow{1} C_2 \xrightarrow{2} 2 \ldots$$

Assume that $P_2(1,2)$ is not contiguous, and since player 1 was the last player to have done best response in reaching C_2 it follows that he did not choose a strategy which results in the shared segment being contiguous in C_2. But now, we use this fact to analyze the last response of player 2, who started from C_0. Since player 1 was able to take shortcuts across segments of player 2's path, we can construct a better response for player 2 starting from C_0, which is a contradiction.

In the inductive step, we have to show that for any configuration C_{i+1}, the edges $P_{i+1}(1,2)$ are contiguous and $P_{i+1}(1,2) \supseteq P_i(1,2)$. The fact that $P_{i+1}(1,2)$ is a contiguous path follows essentially from the same proof as in the base case. Given that, we now have to consider only the strategies in Figure 4.2.

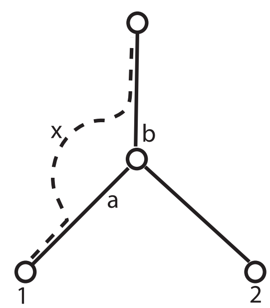

Figure 4.2: The only possible deviations for a two-player game.

Indeed if player 1 decides to take the path as in the figure, taking the shortcut through path x

$$f_x(1) < f_a(1) + f_b(2)/2$$

By inductive hypothesis, the shared part only grew till now, so when when player 1 had last done best response from configuration C_{i-1}, player 2 could not have been on any edges of the subpaths x or a. So it must have been the case that

$$f_x(1) > f_a(1) + f_b(2)/2$$

which is a contradiction. Hence, this is not a valid deviation for player 1. Thus, either $P_{i+1}(1,2) = P_i(1,2)$ or $|P_{i+1}(1,2)| > |P_i(1,2)|$. But note that the two paths $P_i(1) -$

$P_i(1,2)$ and $P_i(2) - P_i(1,2)$ are always shortest paths and so $P_{i+1}(1,2) = P_i(1,2)$ implies we have reached a Nash. Else $P_i(1,2)$ strictly increases by at least one edge. Hence, we reach a Nash in polynomial number of steps. ■

The above proof shows that for any best response run, the number of edges shared by both players increases monotonically. For more players, however, the hope of any positive result about best response dynamics seems slim. In fact, we can show the following.

Theorem 4.5.2 *Best response dynamics for k players may run in time exponential in k.*

To prove this, we now show an example in which by appropriate ordering of the best response of players, we can simulate a k-bit counter.

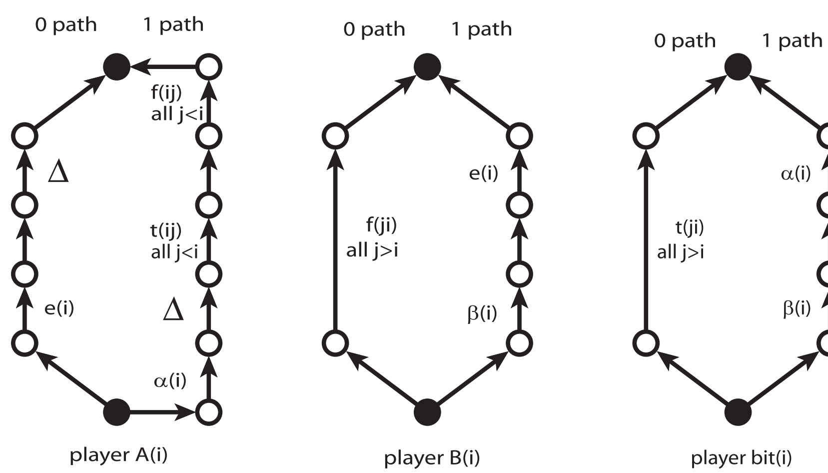

Figure 4.3: The construction of an exponential Best-Response run.

The graph has $3n$ players, n "bit" players, each being assisted by two auxiliary players. The auxiliary players of the i^{th} player are denoted by $A(i)$ and $B(i)$. Each bit player and each auxiliary player has only two path options, we call these the 0 path and the 1 path. Note that the different players share edges, which are identified by the label that is on them. Specifically, the same edge $f(j,i)$ for instance, is present in the 0-path of all $B(i)$ players when $j > i$ and also in the 1 path of all the $A(j)$ players such that $j > i$. We also refer to the player going on the one path as the *player being set* and going on the zero path as the *player being reset*. Each player has one source and one sink and the paths of each player are as shown in the gadgets above. The costs of the paths of i^{th} bit player are referred to as $x_i^{(0)}$ and $x_i^{(1)}$, and those of the player $A(i)$ and $B(i)$ as $a_i^{(0)}$, $a_i^{(1)}$ and $b_i^{(0)}$, $b_i^{(1)}$ respectively. Here we describe how the counter works and the inequalities that should hold for it to work properly.

Start Step : All the players are reset.

General Step : The bits from 1 to $l-1$ are all set. The bits from $l+1$ to n maybe at 0 or 1. The l^{th} bit is currently at 0 and has to be set at 1. Also, all the $A(j)$ players are reset. The $B(j)$ players are set if and only if the j'th players are set.

- Now, the l^{th} bit sets. This triggers both $A(l)$ and $B(l)$.

  At this point, there are no other players using either of bit l player's paths, so for this to happen, we need only the following to be true. In writing the deviation requirements that we need, for clarity's sake, we first write it in a "short" and (arguably) more understandable form. In the next line we expand it in terms of the constituent edges.

$$
\begin{aligned}
x_l^{(1)} &< x_l^{(0)} \\
\alpha(l) + \beta(l) &< \sum_{j>l} t(j,l)
\end{aligned}
$$

- The cost of 1 path of $A(l)$ has now decreased by $\sum_{j<l} c_{lj}/2 + \alpha(l)/2$ because of this bit and other players that were already there. $A(l)$ is triggered and is allowed

to set. Thus,

$$\begin{aligned} a_l^{(1)} - \sum_{j<l} c_{lj}/2 - \alpha(l)/2 &< a_l^{(0)} \\ \alpha(l)/2 + \sum_{j<l} f(l,j) + \sum_{j<l} c(l,j)/2 + \sum_{j<l} t(l,j) &< \sum_{j>l} c(j,l) + e(l) \end{aligned}$$

- The setting of $A(l)$ triggers all the $B(i)$ for $i < l$ to be reset. Recall that the corresponding $A(i)$ are already reset. We allow these $B(i)$ to reset. Due to the triggering, the cost of 0 path has changed by $f_{li}/2$. We also need to take into account that the 1 path is possibly shared by others too.

$$\begin{aligned} b_i^{(0)} - f(l,i)/2 &< b_i^{(1)} - e(i)/2 - \beta(i)/2 \\ \sum_{j>i} f(j,i) - f(l,i)/2 &< e(i)/2 + \beta(i)/2 \end{aligned}$$

- $A(l)$ also triggers all the bits $i < l$ to reset by reducing the cost of the 0 path of bit-i by $t(l,i)/2$. We allow that too.

$$\begin{aligned} x_i^{(0)} - t(l,i)/2 &< x_i^{(1)} \\ \sum_{j>i} t(j,i) - t(l,i)/2 &< \alpha(l) + \beta(l) \end{aligned}$$

- Now because of the setting of the bit-l, the 1 path of B_l became cheaper by $\beta(l)/2$. $B(l)$ wants to set and is allowed to.

$$\begin{aligned} b_l^{(1)} - \beta_l/2 &< b_l^{(0)} \\ e(l) + \beta(l)/2 &< \sum_{j>l} f(j,l) \end{aligned}$$

- Lastly, as a result of the setting of $B(l)$, the 0 path of $A(l)$ became cheaper by $e(l)/2$. $A(l)$ now wants to reset. The 1 path of $A(l)$ is possibly shared by other players.

$$a_l^{(0)} - e(l)/2 \quad < \quad a_l^{(1)} - \sum_{j<l}(f(l,j)/2 + c(l,j)/2 + t(l,j)/2) - \alpha(l)/2$$

$$e(l)/2 + \sum_{j>l} c(j,l) \quad < \quad \alpha(l)/2 + \sum_{j<l}(f(l,j)/2 + c(l,j)/2 + t(l,j)/2)$$

- Now we have the subgame from 1 to $l-1$ being completely reset, and no other player from the top part influencing any of their paths. So we can play their complete game and come back to the configuration in the start of the recursion, except now we need to deal with the $(l+1)^{st}$ bit.

Proof : Theorem 4.5.2 We now prove that the above game has an exponential best response run under the above best response scheduling.

All we need to show is that the moves described in the scheduling are best responses. We first argue that each player has only two possible paths available to him, which we have described as the zero path and the one path. To complete the construction we next need to come up with a set of values for the links that satisfy the set of best response inequalities. Taken together, it follows that the moves are all best responses and hence simulate an exponential length counter.

First we need to argue that each player has only the two path available to him. Intuitively the argument is as follows. If one player, say bit(i), deviates out of his gadget, he will only travel to corresponding bit players with indices growing or decaying monotonically, so that he will never be able to come back to his own gadget. Thus, the only available strategies to a player are the 0 and 1 paths. We illustrate in detail for the i-th bit player. Along $x_i^{(0)}$, the only shared nodes and edges of this player are the $t(j,i)$ edges that are each shared with $A(j)$ for all $j > i$. Suppose this bit player follows the $t(x,i)$ edge to the gadget of $A(x)$ with $x > i$. But from then on he can only travel on edges of the form $t(x, y)$ for $i + 1 \leq y < x$ and will not get a directed path back to the sink of the i^{th} bit player. We can make similar arguments for all the other paths of all players.

For the last part of the construction, we show that it is possible to come up with a set of values for the links such that the best response inequalities are satisfied. Let the edge costs be as follows. In all the remaining formulae ε represents arbitrary small positive numbers less than $\frac{1}{2}$ and c be any constant greater than 5, say. Let $\alpha(1) = 2n - 1 + \varepsilon$, and $\alpha(i) = 2n - 3i + 1 + 3\varepsilon/2$ for $i > 1$. Let $c(i, j) = 1$ for all i and j. Let $\beta(1) = 1$ and $\beta(i) = 2c^i(n - i + \varepsilon - 1/2) - 2e(i)$. Also, $e(1) = 1$ and $e(i) = \sum_{j<i} f(i, j) + \sum_{j<i} t(i, j) + \varepsilon$ for all $i > 1$. Finally, $f(i, 1) = 1$ for all i and $f(i, j) = c^j$ for all pairs i, j with $j > 1$. And $t(i, 1) = 1$ and $t(i, j) = (2 + \varepsilon)c^j + 2j$. Given these values, we can check that the inequalities are satisfied for all $i < n$, and thereby we can have a exponential sized run. ■

4.6 Weighted Players

So far we have assumed that players sharing an edge e pay equal fractions of e's cost. We now consider a game with fixed edge costs where players have weights $w_i \geq 1$, and players' payments are proportional to their weight. More precisely, given a strategy $S = (S_1, \ldots, S_k)$, define W to be the total weight of all players, and let W_e be the sum of the weights of players using e. Then player i's payment for edge e will be $\frac{w_i}{W_e} c_e$.

Note that the potential function $\Phi(S)$ used for the unweighted version of the game is not a potential function once weights are added. In particular, in a weighted game, improving moves can increase the value of $\Phi(S)$, as this is no longer a congestion game. The following theorem uses a new potential function for a special class of weighted games.

Theorem 4.6.1 *In a weighted game where each edge e is in the strategy spaces of at most two players, there exists a potential function for this game, and hence a Nash equilibrium exists.*

Proof : Consider the following potential function. For each edge e used by players i and j, define

$$\Phi_e(S) = \begin{cases} c_e w_i & \text{if player } i \text{ uses } e \text{ in } S \\ c_e w_j & \text{if player } j \text{ uses } e \text{ in } S \\ c_e \theta_{ij} & \text{if both players } i \text{ and } j \text{ use } e \text{ in } S \\ 0 & \text{otherwise} \end{cases}$$

where $\theta_{ij} = (w_i + w_j - \frac{w_i w_j}{w_i + w_j})$. For any edge e with only one player i, simply set $\Phi_e(S) = w_i c_e$ if i uses e and 0 otherwise. Define $\Phi(S) = \sum_e \Phi_e(S)$. We now simply need to argue that if a player makes an improving move, then $\Phi(S)$ decreases. Consider a player i and an edge e that player i joins. If the edge already supported another player j, then i's cost for using e is $c_e \frac{w_i}{w_i + w_j}$, while the change in $\Phi_e(S)$ is

$$c_e(w_i - \frac{w_i w_j}{w_i + w_j}) = c_e \frac{{w_i}^2}{w_i + w_j}.$$

Thus the change in potential when i joins e equals the cost i incurs, scaled up by a factor of w_i. In fact, it is easy to show the more general fact that when player i moves, the change in $\Phi(S)$ is equal to the change in player i's payments scaled up by w_i. This means that improving moves always decrease $\Phi(S)$, thus proving the theorem. ■

Note that this applies not only to paths, but also to the generalized model in which players select subsets from some ground set. The analogous condition is that no ground element appears in the strategy spaces of more than two players.

Corollary 4.6.2 *Any two-player weighted game has a Nash equilibrium.*

While the above potential function also implies a bound on the price of stability, even with only two players this bound is very weak. However, if there are only two players with weights 1 and $w \geq 1$, then we can show that the price of stability is at most $1 + \frac{1}{1+w}$, and this is tight for all w.

The following result shows the existence of Nash equilibria in weighted single commodity games.

Theorem 4.6.3 *For any weighted game in which all players have the same source s and sink t, best response dynamics converges to a Nash equilibrium, and hence Nash equilibria exist.*

Proof : Start with any initial set of strategies S. For every $s - t$ path P define the marginal cost of P to be $c(P) = \sum_{e \in P} \frac{c_e}{W_e}$ where W_e depends on S. Observe that if player i currently uses path P, then i's payment is $w_i c(P)$. Define $P(S)$ to be a tuple of the values $c(P)$ over all paths P, sorted in increasing order. We want to show that the cheapest improving deviation of any player causes $P(S)$ to strictly decrease lexicographically.

Suppose that one of the best moves for player i is to switch paths from P_1 to P_2. Let $\mathcal{P}$ denote the set of paths that intersect $P_1 \cup P_2$. For any pair of paths P and Q, let $c_P(Q)$ denote the new value of $c(Q)$ after player i has switched to path P. To show that $P(S)$ strictly decreases lexicographically, it suffices to show that

$$\min_{P \in \mathcal{P}} c_{P_2}(P) < \min_{P \in \mathcal{P}} c(P). \tag{4.4}$$

Define $P' = \arg\min_{P \in \mathcal{P}} c(P)$. Since P_2 was i's best response, $c_{P_2}(P_2) \leq c_P(P)$ for all paths P. In particular, $c_{P_2}(P_2) \leq c_{P'}(P')$. We also know that $c_{P'}(P') \leq c(P')$, since in deviating to P', player i adds itself to some edges of P'. In fact, $c_{P'}(P') < c(P')$ unless $P' = P_1$. Assuming $P' \neq P_1$, we now have that $c_{P_2}(P_2) < c(P')$, which proves inequality 4.4. If $P' = P_1$, then since player i decided to deviate, $c_{P_2}(P_2) < c(P_1)$. Therefore, we once again have that $c_{P_2}(P_2) < c(P')$, as desired. ■

In the case where the graph consists of only 2 nodes s and t joined by parallel links, we can similarly show that any sequence of improving responses converge to a Nash equilibrium.

With arbitrarily increasing cost functions, [46] gives an example demonstrating that a weighted game may not have any pure Nash equilibria. More recently, Chen and Roughgarden [14] show that our weighted game with fixed cost need not have pure equilibria. The following theorem states that the price of stability bounds from the unweighted case do not carry over even when pure equilibria do exist.

Theorem 4.6.4 *There are weighted games for which the price of stability is* $\Theta(\log W)$ *and* $\Theta(k)$.

An example exhibiting this is a modified version of the graph in Figure 4.1. Change the edge with cost $1+\varepsilon$ to cost 1, and for all other edges with positive cost, set the new cost to be $\frac{1}{2}$. For $1 \leq i \leq k$ let player i have weight $w_i = 2^{i-1}$. Since each player has a greater weight than all smaller weight players combined, the only Nash equilibrium has cost $\frac{k}{2} = \Theta(\log W)$, while the optimal solution has cost 1. See [14] for additional results on weighted games.

Chapter 5

Pricing in Networks

5.1 Introduction

One of the most surprising features of the Internet is how effectively it operates on a global scale, despite the fact that its various components (Autonomous Systems) are operated by separate service providers, each of whom seeks only to maximize their own income. In chapters 3 and 4 we considered competitive network design games as simple models of how selfish agents might construct such a network. These models, as with others in the literature, assume a static user population, typically with fixed demands. In this chapter we address the fact that the potential network users are just as crucial to the construction of a network as the service providers themselves. Network managers compete for users via prices and the quality of service provided. We propose a simple game to explore a basic question that arises in this situation: how is the quality of service affected when the service providers set prices so as to extract maximum profit? We consider the special case of a graph with parallel links (or scheduling parallel machines), where the quality of the service is modeled as delay that increases linearly with the congestion, and user demand is concave. For such a game, we show that a pure equilibrium exists, and we provide a constant factor bound on the price of anarchy. We give an improved bound for the special case in which delay is a pure congestion effect (when delay is 0 with no congestion).

Selfish routing and network design are two important classes of network games that have received much attention in recent years. In work on selfish routing (or load balancing) [17, 20, 21, 45, 61, 60, 63, 66] users in a network route their traffic selfishly with the aim of minimizing their latency. These papers bound the price of anarchy in the games they consider. That is, they give bounds on the performance degradation caused by the selfish routing as compared to a centrally designed optimal solution. In these games the sole selfish goal of users is to minimize the delay; in particular, user demand is fixed (independent of the delay in the system), and the network is passive, in that it does not try to effect routing behavior by changing the properties of its edges.

In this chapter we develop a model that encompasses key aspects of both selfish routing and network design games. Our main result is a small constant bound on the price of anarchy in a game that combines profit-maximizing edge-pricing players with user demand that is sensitive to both prices and congestion. In our model, users perceive the quality of a route (path) in a network via a combination of prices and congestion. Edges have congestion sensitive delays, i.e. the time required to traverse an edge depends on the amount of traffic using it. To this extent, our model extends the work on selfish routing discussed above. However, we will also assume that each edge is operated by a distinct selfish player, who can charge traffic for the use of that edge. Finally, we assume that user demand is affected by the quality of service provided, in that increasing prices and delay lead to decreasing user traffic. We assume that the edges set prices in a selfish manner, aiming to maximize their income. The goal of our work is to quantify the performance degradation caused by selfishness (the price of anarchy) in this model. We show that in the special case of networks with parallel links and linear delays, pure Nash equilibria exist and the price of anarchy is bounded by a small constant.

5.1.1 Model

We define a simple model that combines the issues mentioned above. We consider a network consisting of two nodes s and t, with k parallel links. Each link is controlled by a distinct player, who can charge traffic a price for use of her link. One can also view this game as modeling a type of selfish load balancing problem. Here the edges correspond to machines, and the flow or traffic will selfishly balance between them. Each machine has a load dependent delay, and it can charge a price to all users.

Link i (or machine i) has a latency (or delay) function $\ell_i(x)$, indicating the delay experienced by a volume of x traffic using i. We will primarily focus on strictly increasing linear latencies, i.e. $\ell_i(x) = a_i x + b_i$, where $a_i > 0$ and $b_i \geq 0$. We assume that user experience in routing flow through link i depends on the sum of the price and the latency, which we will call the *disutility*. More precisely, if player i charges p_i and f_i volume of flow uses link i, then that flow experiences a disutility of $p_i + \ell_i(f_i)$.

We assume that traffic routes itself *selfishly*, meaning that traffic will not route along one link if it can switch links and incur a lower disutility. As a result, all traffic will necessarily experience the same disutility.

Finally, we also assume that the total amount of traffic from s to t is dependent on the disutility that traffic experiences; as the disutility increases, the total amount of flow interested in routing from s to t decreases. We model this with a demand function $D(y)$, also referred to as the *demand curve*, which indicates the amount of flow willing to incur a disutility of y. We will naturally assume that demand $D(y)$ is decreasing in disutility y. We will focus on demand curves that are continuous and concave. Different demand curves are used to model demand in different industries. A concave demand curve is applicable for modeling demand for a service with a comparable alternative (at a high enough price all users will switch to the alternate service).

It will often be useful for us to consider $u(x) = D^{-1}(x)$, which we call the *disutility curve*. This measures the disutility that will be tolerated by a volume of x flow. Given our assumptions on $D(x)$, we know that $u(x)$ is also decreasing and concave.

To define our problem more precisely, we say that a price vector p induces a flow vector f satisfying the following properties.

1. For any i, j if $f_i > 0$, then $\ell_i(f_i) + p_i \leq \ell_j(f_j) + p_j$.

2. If $f_i > 0$ then $\ell_i(f_i) + p_i = u(\sum_j f_j)$.

The first condition ensures that no traffic can decrease its disutility by rerouting. The second condition states that the disutility experienced by any traffic must match the the disutility that is tolerated by the given volume of flow. It is not hard to see that since the disutility is continuous and decreasing, such flows always exist. Furthermore, Lemma 5.2.1 will argue that these flows are unique as well.

We may now complete the definition of the game. Each player i selects a price p_i for her link. These prices, together with the latencies and the disutility curve, induce a unique flow f_i on each link. Players seek to maximize their profit, $\pi_i = p_i \cdot f_i$. We say that a set of prices p is at *Nash equilibrium* if, by changing a single price p_i to p'_i, the resulting flow f' does not give player i a larger profit ($\pi'_i = p'_i \cdot f'_i$).

For a set of prices, we are interested in measuring the *social utility* of a given solution. For a solution with prices p inducing total flow $F = \sum_j f_j$, and disutility d, we define the social utility to be

$$U(p) = \sum_i \pi_i + \int_0^F (u(x) - d)dx.$$

The first term accounts for players' profits, and the second terms represents the utility gathered by the traffic that gets routed. An example of an instance of the game with two links is shown in Figure 5.1. The social utility of this instance is indicated by the shaded area.

We will be interested in bounding the *price of anarchy* of this game. If p^* is the set of prices that maximizes social utility, then the *price of anarchy* is the maximum possible

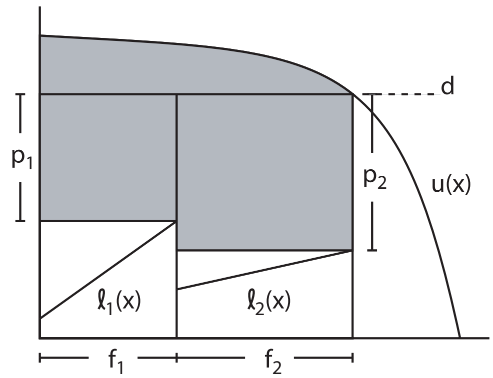

Figure 5.1: An instance of the network pricing game with two players.

ratio of

$$\max_p \frac{U(p^*)}{U(p)},$$

where the prices p range over the possible equilibrium prices. Note that the total area under $u(x)$ provides a trivial upper bound on the maximum possible social utility. However, to achieve this bound we would need to route the maximum possible traffic volume without incurring any delay.

We note that there are other reasonable measures of social ultility; we could have considered either the players' profits or the users' utility as social welfare functions. However, the sum of these two objectives seems the most reasonable, as it captures the total welfare of the system when money is viewed as being fully transferable. Furthermore, under either measure alone, simple examples demonstrate an unbounded price of anarchy.

We can think of our problem as a type of two-stage game where there are two types of players: the owners of the edges, and the users with traffic. In the first stage, the edge-players set prices p_i on edges. In the second stage, traffic routes itself selfishly from the source to the sink with respect to new latencies $\ell_i(x) + p_i$, where the rate of flow is dictated by the disutility curve. For the remainder of this paper, we will focus on the single stage version of this game, the game of setting prices, and will assume that players anticipate the flow resulting from their chosen prices. However, the existence and uniqueness of these flows follow directly from observing this connection.

5.1.2 Results

In this chapter we prove the following results. In section 5.2 we show that for our pricing game, in a network with parallel links, concave demand and linear delays, there is always a pure Nash equilibrium. To reach this result, we first prove a number of lemmas that will be useful throughout this chapter.

In section 5.3 we present the main result of this chapter; the price of anarchy in this game by 4.65. This result holds even when delays are relaxed to be convex, in the case that pure equilibria exist. However, we show in section 5.4 that such pure equilibria might not exist. We also consider a number of variants of this game. We prove that in the special case when delay is exclusively a congestion effect (for all links i, $\ell_i(0) = 0$), the price of anarchy is bounded by at most 3.125. We give a lower bound of 1.5 on the price of anarchy even in the case of a monopolist (i.e. a single player). Lastly, we show that if demand is convex, the price of anarchy may be unbounded.

5.1.3 Related Work

In our model, traffic evaluates its experience via a combination of price and delay. Modelling traffic as experiencing disutility in terms of price and delay has been studied in the transportation literature as early as in 1920 [56] (see also [11]) and more recently in a sequence of papers started by Cole, Dodis and Roughgarden [18, 19] (see also [27, 41]). These works view prices as a tax that is set by a benevolent network manager so as to improve network performance. Acemoglu and Ozdaglar [1] consider a version of our pricing problem in a monopolistic setting with fixed demand. They assume that all edges are owned by a single service provider, and focus on establishing the existence of an equilibrium and characterizing this equilibrium. More recently and independent of this work, the same authors extended their model to oligopolies, while still assuming fixed demand and delays which are exclusively due to congestion, i.e. $\ell_i(0) = 0$ for all i (see [2]). They thereby analyze a special case of our model and prove a tight bound of 6/5 for the price of anarchy. Along different lines, Vetta [67] shows a nice bound on the price of anarchy in a two-stage pricing game for facility location. In Vetta's game the players' strategies are the facilities they select, and prices are determined based on the facility locations, much as our prices determine the flow through the system.

There is a large body of economics literature dedicated to understanding the effects of pricing with service delays, with the focus being on establishing the existence of stable equilibria, and considering qualitative properties of equilibria (such as whether improved service leads to improved profit). Lee and Mason [47] consider the most closely related model with identical links and non-uniform users. More complex models are discussed in the recent papers by Allon and Federgruen [6] and Afèche [3] and their references. Just as in our work, these papers assume that user experience depends on a combination of price and delay, and that prices are set by selfish, income-maximizing players. However, these papers consider more complex environments, including heterogeneous users, and situations in which not only prices, but also delays can be used as strategic variables. However, unlike our work, these papers do not provide bounds on the quality degradation caused by selfish pricing.

5.2 Existence of Pure Equilibria

We start by proving that a given price vector induces a unique flow vector. For the pricing game to be well defined, this simple result is crucial; a unique flow vector implies that players can predict the profit they will extract given any particular set of prices, and thus they are in a position to behave rationally.

Lemma 5.2.1 *For a given price vector p there is a unique flow vector f satisfying conditions 1 and 2 above.*

Proof : For a given value for disutility d, the total amount of flow F can be calculated in two ways. First, the disutility curve gives us the total flow value of $u^{-1}(d)$. Second, on each individual link i, we know that the flow has to be $max(\ell_i^{-1}(d - p_i), 0)$, and hence the total flow must be $\sum_i max(\ell_i^{-1}(d - p_i), 0)$. Since disutility is decreasing, the first function is monotone decreasing in d. Furthermore, since latencies are strictly increasing and continuous, the second function is strictly increasing in d. Hence they can have at most a single intersection point. ■

We say that a player is *content* if she has no incentive to deviate from her current price. To prove that pure Nash equilibria exist, it will be critical to relate the price charged by a content player to the flow she receives, and to the slopes of the latency and disutility functions. This is achieved through the following technical lemma, which will also play a key role in later bounding the price of anarchy.

Lemma 5.2.2 *Let p be the price vector chosen by the players and f be the corresponding flow vector, with total flow $F = \sum_i f_i$. Define v^- and v^+ respectively as the left and right derivatives (slopes) of the disutility curve at F. We will assume that v^- and v^+ are both well defined (note that they are both negative), though they do not have to be equal.*

If player i is content, then the following two conditions must hold:

1. $(\frac{p_i}{f_i} - a_i)(\sum_{j \neq i} \frac{1}{a_j} - \frac{1}{v^-}) \geq 1$

2. $(\frac{p_i}{f_i} - a_i)(\sum_{j \neq i} \frac{1}{a_j} - \frac{1}{v^+}) \leq 1.$

Observe that equality must hold if the disutility curve is continuously differentiable at F.

Before proving this, we establish some basic results concerning the monotonicity and continuity of this game. We first show that if a single player changes her link price, the flow induced by all players' prices changes in the natural way. In particular, if a player increases her price, the newly induced flow will route less traffic on her link, at least as much traffic on every other link, and less traffic in total. Similarly, the symmetric claims hold if the player decreases her price. More precisely we have

Lemma 5.2.3 *Let $p_1, \ldots, p_k$ be a price vector with associated flow vector $f_1, \ldots, f_k$. Assume that the first player increases her price to $p'_1 > p_1$ while the others keep their prices unchanged. Denote the new flow vector by $f'_1, \ldots, f'_k$. If $f_1 > 0$, then,*

1. $f'_1 < f_1$,

2. $f'_i \geq f_i$, *for all* $i \neq 1$.

3. $F' \leq F$, *where* $F = \sum_i f_i$ *and* $F' = \sum_i f'_i$.

The symmetric claim holds if player 1 reduces her price.

Proof : We will assume without loss of generality that there are at least two players with positive flow, since otherwise the claims are trivial.

1) Suppose the flow on link 1 does not decrease. Then, since the price increased, the disutility must strictly increase. Consider any other link that had positive flow. This

link now has greater disutility, and since the price has not changed, must carry more flow. But this implies that both the disutility and total flow have increased, which is a contradiction.

2) Suppose some link $i \neq 1$ loses flow. Since p_i is unchanged and latencies are strictly increasing, the disutility must strictly decrease. This implies that all links that carried flow must now carry less flow. Since we know that link 1 loses flow, the total flow decreases. But then both the disutility and the total flow have decreased, which is a contradiction.

3) Finally, since flow on any link other than 1 can only increase, the disutility can only increase, and thus the total volume of flow can only decrease. ■

The next lemma argues that the amount of flow routed along any link is a continuous function of the players' prices.

Lemma 5.2.4 *Let p be a price vector and let $f_i(p_i)$ be the amount of flow routed through link i as a function of p_i (assuming all other prices in p are fixed). Then the function $f_i(p_i)$ is non-increasing and continuous for any p.*

Proof : The non-increasing part follows from Lemma 5.2.3. As for continuity, we again prove by contradiction. We will argue that if there is a discontinuity, a small change in price can be found that violates monotonicity of disutility. For some price vector p let p_i be a point of discontinuity of $f_i(\cdot)$. Let f be the flow vector corresponding to the prices p with player i charging p_i. Assume without loss of generality that $f_i(\cdot)$ is upper-discontinuous at p_i. Then there exists a small $\Delta > 0$ such that for any $\epsilon > 0$, if player i increases p_i by ϵ, the flow going through the i^{th} link will decrease by at least Δ (notice that it must decrease by Lemma 5.2.3). Since latencies are strictly monotone, the loss of at least Δ flow decreases the latency on link i by at least some $L > 0$. Choose $\epsilon < L$.

Then if player i selects a price of $p_i + \epsilon$, the resulting latency will more than offset the increase in price. Thus the disutility will strictly decrease, contradicting Lemma 5.2.3.

■

Corollary 5.2.5 *The profit $\pi_i(p_i) = p_i \cdot f_i(p_i)$ of player i as a function of the price she chooses, given a fixed price vector p for the other players, is continuous for any p.*

The next lemma argues that the not only the profit of player i, but also that of any other player j changes continuously as i changes her price, while other players (including j) maintain their existing prices.

Lemma 5.2.6 *The profit of any other player j as a function of the price chosen by player i (assuming that the other prices are fixed), is continuous and non-decreasing.*

Proof : The non-deceasing part follows from Lemma 5.2.3. Let $f_i(p_i)$ and $f_j(p_i)$ be the flows of the players i and j as a function of the price p_i chosen by player i. Since $\ell_i(.)$ is continuous and player i's flow continuously depends on p_i (Lemma 5.2.4), $\ell_i(f_i(p_i))+p_i$ also continuously depends on p_i. On the other hand, since both links i and j carry non-zero flow, $\ell_i(f_i(p_i)) + p_i = \ell_j(f_j(p_i)) + p_j$, where p_j is the fixed price of player j. Thus, $\ell_j(f_j(p_i))$ is also continuous in p_i. Furthermore, since $\ell_j(.)$ is strictly monotone, $f_j(p_i)$ must be continuous in p_i. Once the flow on link j is continuous, the continuity of the profit of player j is immediate. ■

Having shown that player profits depend continuously on the price of any player, we can extend this trivially to the following general profit continuity lemma.

Lemma 5.2.7 *Let $\pi : \mathbb{R}^k \mapsto \mathbb{R}^k$ be the function mapping a vector of prices p to the vector of profits obtained by the players when they charge the corresponding prices, i.e.*

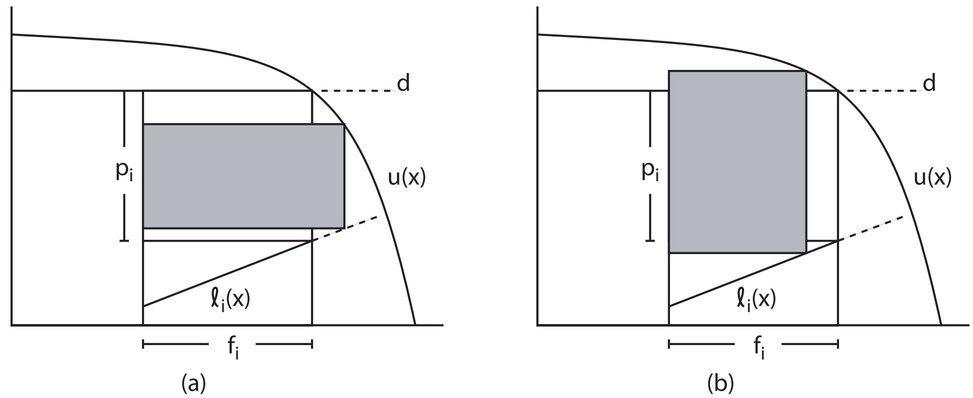

Figure 5.2: The immediate effect of a player (a) decreasing and (b) increasing her price.

$\pi(p) = (\pi_1, \ldots, \pi_k)$ *means that the profit of player* i *is* π_i *under the price vector* p*. Then the function* $\pi(\cdot)$ *is a continuous function of* p*.*

This continuity allows us to prove Lemma 5.2.2.

Proof of Lemma 5.2.2 : The intuition behind the proof of this lemma is based on the following observations. For a player i to be content, she must not be able to benefit by changing her price at all. The lemma will follow from observing that, in particular, she will not benefit from very small changes to her price. Intuitively, the gain from an increase in price must be outweighed by the resulting loss of flow. In Figure 5.2 the shaded boxes indicate the immediate effect of a player increasing or decreasing her price *without* taking into account the resulting change in flow of the other players. The actual resulting flow depends not only on player i's latency function, but also on the disutility curve (determining how much flow leaves the system altogether), and other players' latencies (determining how much of the flow of player i transfers to them; flatter latency players will steal more flow than steeper latency ones). For sufficiently small increases in price, the magnitude of this loss is dictated solely by the local slopes of these curves, resulting in the first inequality. Likewise, the second inequality follows by considering a small decrease in price.

More formally, let us prove the second inequality. The proof for the first one is analogous. Assume that player i changes her price so that the system disutility is decreased by a tiny amount $\delta > 0$. Then the flow of a player $j \neq i$ will decrease by δ/a_j and the total flow in the system will increasey by $-\delta/v^+$ [1]. Hence player i's flow must increase by $\Delta f_i = \delta(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+})$. This increase in flow will result to a latency increase of $a_i \cdot \delta(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+})$. So i must decrease her price by $\Delta p_i = \delta + a_i \cdot \delta(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+})$.

We next observe that for any $\epsilon > 0$, there exists a small enough δ such that, $\Delta p_i/\Delta f_i \geq p_i/f_i - \epsilon$. Indeed, if $\Delta p_i/\Delta f_i < p_i/f_i - \epsilon$ then $p_i \Delta f_i > f_i \Delta p_i + f_i \Delta f_i \epsilon$. Now player i can obtain more profit by setting her price to $p_i - \Delta p_i$, because her new profit would be $(p_i - \Delta p_i)(f_i + \Delta f_i) = p_i f_i - f_i \Delta p_i + p_i \Delta f_i - \Delta p_i \Delta f_i > p_i \cdot f_i$ if we choose δ small enough to ensure that $\epsilon f_i > \Delta p_i$ (which we can do due to continuity of latencies and disutility). But this can not happen, since player i was content. Thus in the limit of δ approaching 0 we get $\Delta p_i/\Delta f_i \geq p_i/f_i$.

Therefore as δ tends to 0,

$$\begin{aligned}
\delta + a_i \cdot \delta(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+}) = \Delta p_i \geq \frac{p_i}{f_i}\Delta f_i = \frac{p_i}{f_i} \cdot \delta(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+}), \\
1 + a_i(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+}) \geq \frac{p_i}{f_i}(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+}), \\
(\frac{p_i}{f_i} - a_i)(\sum_{j\neq i} \frac{1}{a_j} - \frac{1}{v^+}) \leq 1.
\end{aligned}$$

■

We can now analyze the best response function $\beta : \mathbb{R}^k \mapsto \mathbb{R}^k$. This function maps a price vector p to another price vector p', such that p'_i maximizes player i's profit assuming all other players price as in p. We define player i's best response to be $p_i = 0$ if there is no price at which i can derive a positive profit. We will use Lemma 5.2.2 to show that a player's best response is unique, and hence $\beta(p)$ is well defined. This and

[1] To be formal, we have to take a sequence of δ's approaching 0.

the continuity of $\beta(\cdot)$, which will be shown in Lemma 5.2.9, will be sufficient to prove the existence of pure equilibria.

Lemma 5.2.8 *For any set of existing prices p, all players' best response is unique (and hence the $\beta(p)$ is well defined).*

Proof : Suppose that the best response for some player i is not unique, and let $p'_i > p''_i$ be two best response prices for player i. Let f' and f'' be the corresponding flow vectors when player i selects prices p'_i and p''_i respectively, with corresponding total flow F' and F''. Lemma 5.2.3 implies that $f'_i < f''_i$ and $F' < F''$. Let v' denote the right slope of $u(\cdot)$ at F' and v'' denote left slope of $u(\cdot)$ at F''. By concavity of $u(\cdot)$, $v' \geq v''$ (recall that both values are negative).

Since both p_i and p''_i are best responses for player i, we can apply Lemma 5.2.2:

$$1 \geq (\frac{p'_i}{f'_i} - a_i)(\sum_{j \neq i} \frac{1}{a_j} - \frac{1}{v'}) > (\frac{p''_i}{f''_i} - a_i)(\sum_{j \neq i} \frac{1}{a_j} - \frac{1}{v''}) \geq 1.$$

This is a contradiction, and thus a players best response must be unique. ■

Lemma 5.2.9 *The best response function $\beta(\cdot)$ continuously depends on the price vector.*

Proof : It is sufficient to argue that $\beta(\cdot)$ is continuous for each player in each coordinate of the price vector. Assume not. Let $p = (p_1, \ldots, p_n)$ be a price vector for which player j's best response changes discontinuously when player i increases her price by an arbitrary small amount ϵ. Let $p' = (p_1, \ldots, p_{i-1}, p_i + \epsilon, p_{i+1}, \ldots, p_n)$ denote the new price vector, and $\beta_j(p)$ and $\beta_j(p')$ be respectively the best response prices of player j under the two price vectors. By Lemma 5.2.8, the best response is unique, so the profit that j gets from charging $\beta_j(p)$ under price vector p is strictly greater (say by Δ) than the profit that it would get by charging $\beta_j(p')$ instead. But we know from Lemma 5.2.7 that profit changes continuously in prices. Hence for a small enough ϵ, the profit difference could not reach Δ, contradicting the discontinuity assumption. ■

Since there is a maximum price P such that any player charging above P gathers no profit, we can restrict our attention on $\beta(\cdot)$ to the convex and compact region $[0, P]^k$. Thus we can apply Brouwer's fixed point theorem and thereby prove the following.

Theorem 5.2.10 *The network pricing game has a Nash equilibrium.*

5.3 Price of Anarchy

In this section we prove our main result, namely that selfish behavior on the part of the players yields a social utility that is within a small constant factor of optimal. More formally, we prove

Theorem 5.3.1 *The price of anarchy for the network pricing game is at most 4.65.*

The first step is to take a general instance of a game with prices at Nash equilibrium, and create a new game instance in which the equilibrium gathers no traffic utility (see Figure 5.3). In doing this we preserve the original equilibrium, and only increase the price of anarchy. Now, in this modified game, we can bound the social utility of an optimal solution solely against the player profit at Nash equilibria.

We will consider a general instance of the game with disutility function u and latencies $\ell_i(\cdot)$ for $1 \leq i \leq k$. We assume we have a price vector p at Nash equilibrium, with induced flow vector f, total flow volume F and disutility d.

Lemma 5.3.2 *Define a new disutility curve*

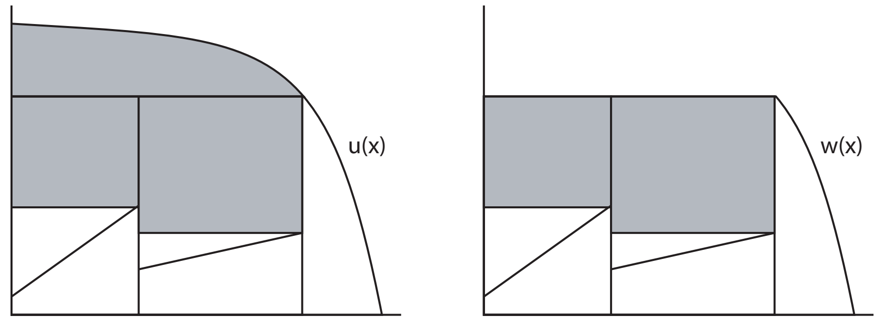

Figure 5.3: A disutility curve with prices and flows at equilibrium and the corresponding truncated curve.

$$w(x) = \begin{cases} d & \textit{if } x < F \\ u(x) & \textit{otherwise} \end{cases}$$

Then prices p are also at Nash equilibrium given this truncated disutility curve, and the price of anarchy of this instance has not decreased.

Proof : Clearly no player has an incentive to decrease her price, as this would increase the total flow, and thus yield a flow vector that was achievable under $u(x)$. But we assumed that p was at equilibrium for $u(x)$, so this can not benefit any player. Furthermore, no player has in incentive to increase their price, as the potential gain of any such move is strictly smaller than it would have been prior to the truncation.

As for the price of anarchy, note that this truncation destroys all traffic utility of the given equilibrium, as shown in Figure 5.3. But this is also an upper bound on the decrease in the social utility of the new optimal solution under the truncated disutility curve. Thus the price of anarchy can only increase. ■

The next lemma provides a simple lower bound on the price that players charge at equilibrium, both in terms of the system disutility d and their own latency function. This result will clearly be useful in lower bounding player profits in the Nash equilibrium.

Lemma 5.3.3 *At equilibrium, any player i charges $p_i \geq \frac{d-b_i}{2}$, where $b_i = \ell_i(0)$.*

Proof : For simplicity, we will assume we have a truncated disutility curve, although the lemma is also true without this assumption. Define $q = d - b_i$. We will claim that if player i charged $p_i < \frac{q}{2}$, then she can increase her profit by charging $\frac{q}{2}$. Observe that $a_i \cdot f_i + b_i + p_i = d$. Thus we can express

$$\pi_i = p_i \cdot f_i = p_i(q - p_i)/a_i < \frac{q^2}{4a_i}.$$

However, since we are assuming a truncated disutility curve, in charging $\frac{q}{2}$, the resulting flow f' would be determined solely by $\ell_i(x)$. Hence, f' satisfies the same condition, i.e. $a_i \cdot f'_i + b_i + \frac{q}{2} = d$. Thus the profit would be exactly $\frac{q^2}{4a_i}$. ■

We now prove our main result. We consider an instance of our game with equilibrium prices p, and corresponding flow vector f, total flow F, and disutility d. We assume that our disutility function $u(x)$ is truncated as in Lemma 5.3.2, i.e. $u(x) = d$ on the interval $[0, F]$, and thus $U(p)$ is represented entirely by player profit.

Proof of Theorem 5.3.1 : As observed, in our truncated instance of the game, the given Nash equilibrium generates no traffic utility. Thus we must bound both the player profit and traffic utility of the optimal solution against the profit of the players at Nash equilibrium. We will consider an optimal price vector p^*, with corresponding flows f^*, F^*, and disutility value d^*. We can trivially bound the traffic utility of the optimal solution by $F^* \cdot (d - d^*)$, and we can attribute $f_i^* \cdot (d - d^*)$ of this bound to each player i. Thus we will think of each player i in the optimal solution as contributing $\pi_i^* + f_i^* \cdot (d - d^*)$ to $U(p^*)$.

We first partition the players by the slope of their latencies. More precisely, let us call a player *steep* if $a_i \geq \frac{1}{2}\frac{p_i}{f_i}$. Otherwise, we will say a player is *shallow*. We will now argue that there is at most one shallow player. Then we will show that the contribution

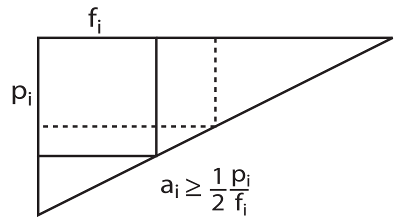

Figure 5.4: Bounding the value that optimum gathers from steep players.

of a steep player i to the optimal solution can be bounded in terms of π_i. Finally, we consider two cases regarding the shallow player's latency. In both cases, we show that the contribution of this shallow player to the optimal solution can be bounded within a constant factor of the sum of the profits of all the players at Nash equilibrium.

Intuitively, the reason why there is only a single shallow player is that if there were two (or more), then one of them would have an incentive to reduce her price slightly, since the amount of extra flow she gains by stealing from the other shallow player(s) would far outweight the slight decrease in her price. More formally, let v^+ be the right slope of $u(x)$ at F. Consider any shallow player i. By Lemma 5.2.2, we know that $(\frac{p_i}{f_i} - a_i)(\sum_{j \neq i} \frac{1}{a_j} - \frac{1}{v^+}) \leq 1$. Since i is shallow, the first term is greater than $\frac{p_i}{2f_i}$. But v^+ is negative, so for all $j \neq i$ we must have $a_j > \frac{p_i}{2f_i} > a_i$, as otherwise the above inequality would be violated. This implies that i must have the unique minimum slope, and thus i must be the only shallow player.

Consider a steep player j. The optimal solution can not gather more value than π_j by charging more than p_j, as then the player could selfishly do the same. Thus we only need to consider how the optimal solution might benefit by having $p_j^* < p_j$. By assumption, $a_j \geq \frac{1}{2}\frac{p_j}{f_j}$. The maximum feasible rectangle (corresponding to gathered utility) that is bounded above by $u(x)$ and below by $\ell_j(x)$ has an area of $1.125\pi_j$, which is achieved when $a_j = \frac{1}{2}\frac{p_j}{f_j}$ by setting a price of $\frac{3}{4}p_j$ and inducing a flow of $\frac{3}{2}f_j$ (see Figure 5.4). Thus nearly all players gather almost as much utility as they would in an optimal solution.

Now we are left with the task of bounding the value gathered by the shallow player i in the optimal solution. We again know that to gather more value with i, the optimal solution must charge a lower price and carry more flow, as any benefit in raising the price could also be realized by the player at equilibrium. Thus our concern is that somehow the optimal solution sends a huge amount of flow at lower price on i, thereby gathering substantially larger social utility. We consider two cases, based on just how shallow i's latency is.

Case 1: $\ell_i(4f_i) - \ell_i(f_i) \geq p_i/4$. In this case the latency is not very shallow, and we can simply bound the maximum contribution of player i to the optimal solution by ignoring all other players and assuming $u(x) = d$. Then the best choice for i is to charge $\frac{d-b_i}{2}$ (thereby maximizing the area of a rectangle inscribed in a triangle). The maximum area of such a rectangle would be when $\ell_i(4f_i) - \ell_i(f_i) = p_i/4$, in which case $\frac{d-b_i}{2} = \frac{13}{24}p_i$ and the corresponding flow is $\frac{13}{2}f_i$ for a total profit of $3.52\pi_i$.

Case 2: $\ell_i(4f_i) - \ell_i(f_i) < p_i/4$. This case deals with a very shallow latency. A trivial upper bound on the amount that the optimal solution can gather through i is

$$\int_0^r (u(x) - \ell_i(f_i))dx$$

where r is defined by $u(r) = \ell_i(f_i)$. This is the value i could gather in the absence of other players if $\ell_i(x)$ never exceeded $\ell_i(f_i)$. We will partition this area into two regions; A, representing $\int_F^r (u(x) - \ell_i(f_i))dx$, and B, representing the rest, with area $F(d - \ell_i(f_i))$, as shown in Figure 5.5(a).

To bound A, recall that $a_i < \frac{1}{2}\frac{p_i}{f_i}$. Since the slope of i's latency is shallow, the disutility curve must be relatively steep, as otherwise, i could decrease p_i slightly and dramatically increase f_i. More precisely, Lemma 5.2.2 implies that $v^+ \leq -\frac{1}{2}\frac{p_i}{f_i}$. Since

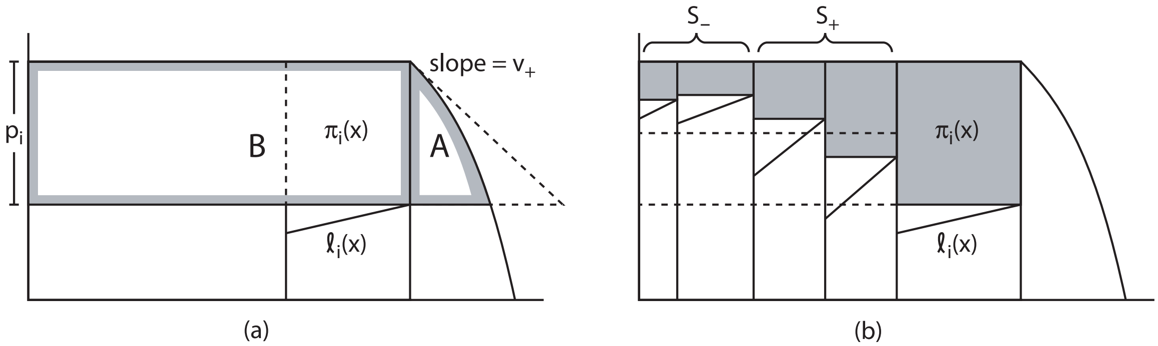

Figure 5.5: (a) The regions A and B for the shallow player, (b) Partitioning the steep players into S^- and S^+.

$u(x)$ is concave, the area in A can be upper bounded by a triangle of height p_i and slope v^+. This has area at most $p_i \cdot f_i = \pi_i$.

To bound B, we partition the steep players (all $j \neq i$) into two classes. Let S^- be the set of all players j for whom $b_j > \ell_i(f_i) + p_i/2$, and let S^+ consist of all remaining steep players, as shown in Figure 5.5(b). Define F^- and F^+ to be the total flow carried by all players in S^- and S^+ respectively at equilibrium. By Lemma 5.3.3, we know that for any player $j \in S^+$, $p_j \geq (d - b_j)/2$, and hence the total profit of all players in S^+ is at least $p_i F^+/4$. Furthermore, we can argue that S^- must be small, as if it was very large, player i would have an incentive to undercut all the players in S^-. In particular, we claim that $F^- \leq 3f_i$. Otherwise, player i would have an incentive to charge a quarter of her current price. Due to the above shallowness condition on her latency, she is guarranteed to more than quadruple her flow before any player in S^- routes any traffic. Clearly this would generate more profit, contradicting our assumption of equilibrium. Thus we can bound the total area of B by $4\pi_i + 4\sum_{j\neq i} \pi_j$.

The combined area of A and B is thus at most $5\pi_i + 4\sum_{j\neq i} \pi_j$, and hence the optimal solution can gather a value of no more than $5\pi_i + 5.125\sum_{j\neq i} \pi_j$. Thus the price of anarchy is at most 5.125.

The promised bound of 4.65 is obtained by rebalancing the different bounds obtained for the different cases in the above proof. First, observe that when the slope of the shallow player was not very shallow (*Case 1* of the above proof), the bound we obtained was 3.52, which is significantly smaller 5.125. Furthermore, returning to our bound on the area of B, note that we can generalize our definition of S^- to be the set of all players j for whom $b_j > \ell_i(f_i) + \alpha p_i$ for some real α, and define S^+ similarly. Parametrizing over the slope of the shallow player a used to distinguish between *Case 1* and *Case 2*, as well as the α, we can try to minimize the maximum of the bounds obtained for the different cases. In particular, if we set $a = 0.06$ and $\alpha = 0.43$, we get the desired bound of 4.65 for the price of anarchy[2]. ■

For the special case when $\ell_i(0) = 0$ for all i, a proof similar to the one above yields the following

Theorem 5.3.4 *The price of anarchy of the network pricing game with a concave disutility curve and linear latencies of the form $\ell_i(x) = a_i x$ is bounded by 3.125.*

Proof : The argument will mostly follow along the lines of the proof of Theorem 5.3.1. As before, there will be at most a single shallow player i and we will bound the value gathered by the optimal solution from the steep players by $1.125 \sum_{j \neq i} \pi_j$. As for the shallow player, we will again split her potential profit into areas A and B. Again, the area of A is at most π_i, but unlike in the proof of Theorem 5.3.1, we can give a simpler bound for the area B. Indeed, Lemma 5.3.3 implies that the area of B is at most

[2]We also need to observe that instead of area A, we can use the smaller area of $\int_F^r (u(x) - \ell_i(f_i + x - F))dx$ to bound the utility that optimum can gather from the shallow player i. Identical to the argument in the proof of Observation 5.3.6, this area is at most $\frac{\pi_i}{2}$, instead of the previous bound of π_i for the area of A.

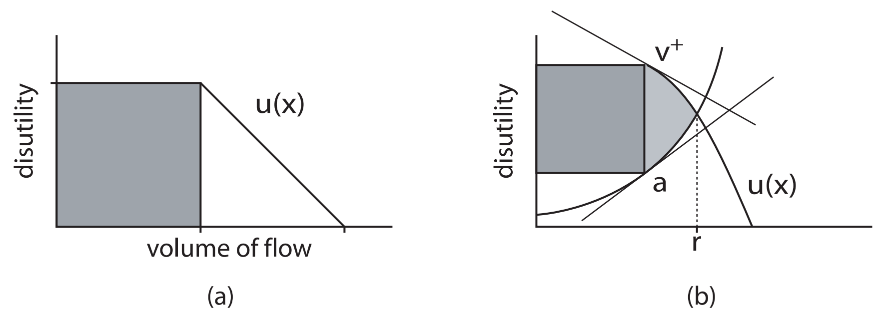

Figure 5.6: (a) A single player example with a price of anarchy of 3/2, (b) Proving the 3/2 upper bound on the price of anarchy of the network pricing game with a single player.

$\pi_i + 2\sum_{j\neq i}\pi_j$, since each of the steep players charges a price of at least $\frac{d}{2}$. Hence the total cost is at most $3.125\sum_{j\neq i}\pi_j + 2\pi_i$ giving the desired guarantee. ■

We conclude the section by presenting a single player instance of our game where the price of anarchy is 3/2. We then show that this bound it tight when there is only a single player by demonstrating a matching upper bound.

Observation 5.3.5 *There exists a 1-player instance of the network pricing game with linear latencies and concave disutility curve which has a price of anarchy of 3/2.*

Proof : Consider the disutility curve $u(x) = 1$ for $0 \leq x \leq 1$ and $u(x) = 2 - x$ for $1 \leq x \leq 2$ and let the player have zero latency (see Figure 5.6(a)). Then she would obtain maximal profit of 1 by charging a price of 1 for a social value of 1. Yet the optimal solution can gather a social value of 3/2 by charging 0. Hence the lower bound. Finally, it is not difficult to slightly modify this example to ensure strict monotonicity of latency curves. ■

Observation 5.3.6 *The network pricing game with a single player has a price of anarchy of at most 3/2.*

Proof : As in the proof of the main theorem, we shall assume truncated disutility curves. We will also assume without loss of generality that $p = f = 1$ for the single player (we can achieve this by rescaling the axes). As before, the only way that the optimal solution can obtain more social value than the equilibrium is by charging a lower price. Hence a trivial upper bound on the social value of the optimum is $1 + \int_1^r (u(x) - \ell(x))dx$, where r is the intersection point of $u(\cdot)$ and $\ell(\cdot)$ (see Figure 5.6(b)). The area of the integral can be bounded by the triangular area bounded by the slopes a and v^+ of the latency and disutility curves. Lemma 5.2.2 tells us that $a - v^+ \leq 1$ (recall that v^+ was negative), so the area of the triangle is at most 1/2. ■

5.4 Extensions and Related Models

In this section we analyze what happens to the network pricing game when we relax the assumptions on the latency functions and the disutility curve. First, we consider convex, as opposed to linear, latency functions, while retaining concave disutility.

Unfortunately, the network pricing game with convex latencies and concave disutility curve may not have a Nash equilibrium as illustrated by the following 2-player example. Let the disutility curve be $u(x) = 1$. Define the player latencies as follows.

$$\ell_1(x) = \begin{cases} 0 & 0 \leq x < 1 \\ \infty & x \geq 1 \end{cases} \qquad \ell_2(x) = \begin{cases} 0 & 0 \leq x < \frac{1}{3} \\ \infty & x \geq \frac{1}{3} \end{cases}$$

We claim that this instance of the game has no Nash equilibrium. Indeed, at an equilibrium the first player can not obtain profit from the first third of the flow, since

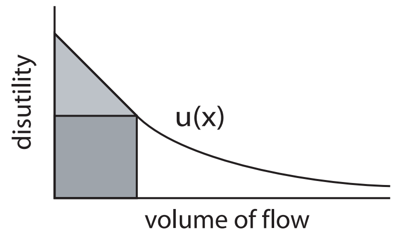

Figure 5.7: An example with convex disutility and unbounded price of anarchy.

otherwise the second player would undercut him. Hence first player's best option must be to charge a price of 1 for a profit of $\frac{2}{3}$. Yet this would induce the second player to charge a high price as well, which in turn would create an incentive for the first player to undercut the second one slightly and obtain a larger profit. Although the above example violates continuity and strict monotonicity of the latency functions and the disutility curve, it is not difficult to alter it slightly so as to satisfy these conditions while still maintaining the nonexistence of Nash equilibrium.

On the other hand, when such a network pricing game does have a Nash equilibrium, the proofs of the previous sections can be extended to yield the same bounds on the price of anarchy.

Theorem 5.4.1 *If an instance of the network pricing game with convex latencies and concave disutility has a Nash equilibrium, then the price of anarchy is bounded by 4.65. Furthermore, if the delay is exclusively due to congestion (i.e. all links have $\ell_i(0) = 0$), then the bound can be improved to 3.125.*

We next consider instances of the game with convex, instead of concave, disutility curves, while maintaining convex latencies. An example similar to the one above can be constructed to show that this game may still not have a Nash equilibrium. Unfortunately, unlike the game with concave disutility curves, even when Nash equilibria do exist, we do not have any such bounds, since as the following single player example illustrates, there exist instances of the game with unbounded gap between the social utility of a Nash equilibrium and an optimal solution.

Example : Consider an instance of the game with a single player who has zero latency, and a disutility curve $u(x) = 2 - x$ when $0 \leq x \leq 1$ and $u(x) = \frac{1}{x}$ otherwise (see Figure 5.7). The equilibrium strategy for the player is to charge any price above 1 for a profit of 1 and finite social utility, yet charging a price of 0 yields and infinite social benefit. ∎

Chapter 6

The Effects of Collusion

6.1 Introduction

The vast majority of research on the price of anarchy, including the previous chapters, considers games in which players act completely independently, with no coordination whatsoever. This is a natural assumption, especially when considering that these games primarily use Nash equilibria as their stable solution concept. By its very definition, a Nash equilibrium models distinct agents who do not work together. Thus in studying the price of anarchy, we are effectively studying the difference between those solutions that can be attained in a fully coordinated manner and those that arise in an entirely uncoordinated fashion. But what about solutions that develop with limited coordination? In many competitive settings, agents do not coordinate fully, but they do attempt to find mutually beneficial arrangements with each other, at least on a small scale. Thus agents may collude to a limted extent, and it is then the coalitions that they form which behave in a self-interested manner. How does such limited collusion effect the quality of the resulting equilibria?

Intuitively, we might expect that forming coalitions should only help the quality of the resulting outcome: at least the members of the coalition are cooperating for a better collective outcome, and overall, the level of coordination in the game only increases. At the extreme, if all players were to form a single coalition, they could reach an optimal outcome that minimizes the total cost for all players. So clearly, coalitions hold the possibility of improvement. But is it possible that under certain settings, the equilibria reached by players in smaller coalitions are in fact more costly than those achieved by players who do not cooperate at all? And if so, to what extent could this happen?

In this chapter, we investigate this question in the context of a very simple class of symmetric congestion games. Congestion games are one of the most well understood and well studies class of games in the context of the price of anarchy. General congestion games were introduced by Rosenthal in [58] as a broad class of games possessing pure equilibria, and were shown to be equivalent to the class of potential games [52]. They are also well understood in the context of computing Nash equilibria [24], and the price of stability [7].

We present a framework for modeling coalitions, and show that, surprisingly, collusion may adversely effect the price of anarchy in a particular game. Our main result is that this increase is bounded by a factor of two if the latencies are convex, and this bound is tight.

6.1.1 Model

We consider a symmetric congestion game with n players and m links. Each player controls a single unit of flow, and must select a *single* link on which to route that flow. Each link s has a non-decreasing latency function $\ell_s(x)$, indicating the delay experienced by players using link s, assuming there are x such players. A pure strategy for player i is denoted $A_i \in [m]$, and a pure strategy $\mathcal{A} = (A_1, \ldots, A_n)$ indicates a strategy for every player. Whenever possible we will prefer pure strategy (deterministic) equilibria, since in many settings, they reflect natural game play more accurately than mixed equilibria. For a given pure strategy $\mathcal{A}$, we define n_s to be the number of players using link s. Suppose under strategy $\mathcal{A}$, player i selects link s (i.e. $A_i = s$). Then we define player i's cost under $\mathcal{A}$ to be $c_i(\mathcal{A}) = \ell_s(n_s)$.

We will also consider mixed (randomized) strategies, where each player chooses a probability distribution S_i on strategies, rather than a single strategy A_i. For a mixed strategy $\mathcal{S} = (S_1, \ldots, S_n)$ we evaluate player i's cost under $\mathcal{S}$ as the expected cost for the player when strategies are selected according to these distributions. We will use $c_i(\mathcal{S})$ to denote this expectation.

A strategy is a Nash equilibrium if no player can lower their cost (or expected cost) by unilaterally deviating to an alternate (mixed) strategy. More formally, a set of pure strategies $\mathcal{A}$ is a Nash equilibrium if for every player i and every link s, $c_i(\mathcal{A}) \leq c_i(\mathcal{A}_{-i}, s)$, where $(\mathcal{A}_{-i}, s) = (\mathcal{A}_1, \ldots, A_{i-1}, s, A_{i+1}, \ldots, A_n)$. An analogous definition can be given for mixed Nash equilibria.

The social cost of a pure strategy $\mathcal{A}$ is simply the sum of all players' costs, $c(\mathcal{A}) = \sum_i c_i(\mathcal{A})$. Equivalently, we can express the social cost as a sum over links, $c(\mathcal{A}) = \sum_s n_s \ell_s(n_s)$. For a game G, if NE is the most costly Nash equilibria and OPT is the solution that minimizes cost, then we define the price of anarchy of G to be $PA = \frac{c(NE)}{c(OPT)}$.

We now present a general framework for modeling static coalitions. We assume that members of a coalition display full cooperation, and aim to minimize their total collective cost. For a game G, we model a given set of coalitions as a partition $P = (P_1, \ldots, P_k)$ of the set of players. We now create a new game, denoted $G(P)$, in which each coalition P_i of players from G is modeled as a single player i, who has to choose a strategy for each of the original players in the set P_i. In the case of our congestion game, coalition i now controls $|P_i|$ units, and must assign each of these units to one of the m links. The goal of coalition i is to minimize the total cost experienced by the units in P_i. Observe that the there is a natural one-to-one correspondence between strategies in the collusion-free game and those in the coalition game. Furthermore, corresponding strategies have the same social cost.

For concreteness, consider the following two extreme cases. If P is the partition of players into n singleton sets, then each coalition consists of a single player, and $G(P) = G$. If P groups all players together, then $G(P)$ is a game containing a single player, whose cost function is the social cost function.

With coalitions defined, we consider Nash equilibria in the game where players represent coalitions. At equilibrium, no coalition has an incentive to deviate to any other assignment of units to links. Note that coalitions have a much richer set of deviations available to them than are available to players in the collusion-free game, as a coalition considers reassigning multiple units simultaneously. Coalitions also evaluate moves differently, as they consider tradeoffs between the delays their members experience. It is not hard to see that a Nash equilibrium in G may not correspond to an equilibrium in $G(P)$, and vice-versa.

Our primary goal is to explore how coalitions can effect the cost of equilibria. More precisely, we are interested in the following question: if G is a collusion-free game, and then coalitions P are formed, yielding the game $G(P)$, how much more expensive can the equilibria in $G(P)$ be (assuming any exist at all) compared to those in G? Let NE be the most costly Nash equilibrium of G (for the games we consider, the Nash equilibrium will be unique under mild technical assumptions). Let CE be the most costly equilibrium over all games $G(P)$ generated by adding any coalitions P to G. We define the *price of collusion* for G to be $\frac{c(CE)}{c(NE)}$, the factor by which the quality of equilibria can deteriorate when coalitions form. For any subclass of games where we have bounds on the price of anarchy (such as classes with restricted kinds of latency functions), we can use the price of collusion to bound the quality of Nash equilibria with arbitrary coalitions.

Notice that our game models *static* coalitions, and we do not model the process of forming coalitions. We assume that coalitions are formed exogenously, and not through game play. We note that while dynamic coalitions are clearly of interest as well, a suitable model has thus far proved elusive.

6.1.2 Results

Our main result is a bound on the price of collusion for simple congestion games with convex or concave latencies. Convex latencies are often used to model delay, or response time, as a function of congestion or load. In section 6.2 we show that if latencies are convex, pure Nash equilibria always exist regardless of how coalitions are formed, even though the resulting game may not be a congestion game.

In section 6.3, we present our main result. For simple congestion games with convex latencies we consider pure equilibria, and show that the price of collusion is most two. We also present a sequence of games with two links for which the price of collusion tends towards two as the number of players increases.

In section 6.4 we consider the case of concave latencies. While our simple congestion games always have pure Nash equilibria, after forming coalitions, the resulting game is not known to have pure equilibria when latencies are concave. We show that the price of collusion is again at most 2 for pure equilibria (if they exist). We also consider mixed equilibria, and show a somewhat weaker bound of 4 on the price of collusion. This proof compares any randomized Nash equilibrium with coalitions to a pure equilibrium of the basic game. We present a weaker lower bound, showing that the price of collusion is at least $8/7$.

For any subclass of these games (e.g., simple congestion games with bounded-degree polynomial latencies), we conclude that coalitions cause the price of anarchy to increase by at most the above small factor.

6.1.3 Related Work

The price of anarchy was introduced by Koutsoupias and Papadimitriou in [45, 54] and first applied to bound the cost of selfish behavior in a simple load-balancing game. This game is very similar to the basic congestion game we consider, with a few notable distinctions; their game considers more general weighted jobs (as opposed to unit jobs), but assumes linear latency functions (we consider general convex and concave latencies), and analyzes the worst case delays of randomized Nash equilibria. See chapter 2 for a discussion of other related load-balancing and congestion games, and for an overview of potential games in general. Note that in contrast to the potential game studied in chapter 4, the underlying games we consider in this chapter have general non-decreasing costs. Furthermore, we consider only singleton strategies, whereas players in the fair connection game could select sets of edges.

The basic game we consider is an unweighted load-balancing game, and hence a congestion game. Many extensions of congestion games have been studied, where, as in our game with coalitions, pure Nash equilibria are guaranteed to exists. Perhaps the best understood are weighted load balancing games, where each job has a weight, and the delay of a machine depends on the total weight (and not the total number) of jobs assigned to it. Such weighted games do not have exact potential functions, yet they do have pure Nash equilibria. Milchtaich [49] considers a generalization of load balancing games in which players have different payoff functions. As in the weighted load balancing games, the result of this extension is not a potential game, and yet the author proves that such games have pure Nash equilibria.

None of the work discussed above considers collusion or its effects. In fact, there is surprisingly little research on the topic in general. We can view the atomic and splittable flow considered by Catoni and Pallottino [12] and Roughgarden and Tardos [63, 62] as a type of coalition similar to those that we consider. These papers consider a splittable atomic routing game in which players control a tangible volume of flow, and may route this flow fractionally over multiple paths. This is the natural version of our model of static coalition formation for a game on a continuum of players that form a finite set of coalitions. In [12], Catoni and Pallottino show an instance of a network with two commodities, each of which chooses between a private and shared path, where the value of splittable atomic equilibrium is worse than the value of the nonatomic Nash equilibrium. In [63] and [62], the authors show that the nonatomic price of anarchy bounds of [63] and [61] respectively also hold for games in which flow is atomic yet splittable. In other words, collusion does not hurt the maximum price of anarchy for the classes of games considered. There are no bounds known on how much collusion can hurt the price of anarchy when it does make it worse.

While there has been a lot of work in the social science literature trying to understand the process of coalition formation, there is no generally accepted game theoretic model of such a process. Explicit study of coalitions in the algorithmic game theory literature to date has been primarily from the perspective of mechanism design. Typically the goal is one of prevention; the designer seeks to create a mechanism in which coalitions have no incentive to form. These mechanisms rely on a central trusted resource, and offer side payments to the agents, with the goal of inducing them to truthfully declare their valuations. For example, a number of papers have dealt with the problem of group-strategyproof cost sharing [53], while Goldberg and Hartline [30] consider group-strategyproof mechanisms in the context of auction design. In contrast, our work does not rely on mechanisms administered by a trusted party, and instead studies the equilibria that result from natural game play. We aim to understand the possible negative effect of coalitions on the outcome of selfish interactions, rather than use a centrally run mechanism to prevent collusion.

6.2 Existence of Pure Equilibria

In this section we show that if any coalitions P are added to a simple congestion game G with convex latencies, then the resulting game $G(P)$ has a pure Nash equilibrium. Recall that G itself is an exact potential game, and thus always has a pure equilibrium. However, there exist simple examples showing that $G(P)$ is not necessarily an exact potential game. Furthermore, there are potential games for which no pure equilibria exist after coalitions are added (see the extended version of this paper.).

The proof of existence is similar to Milchtaich's proof [49] that load-balancing games with user-specific playoff functions have pure Nash equilibria. Let G be a simple congestion game and $P = (P_1, \ldots, P_k)$ describe the k coalitions. We start by considering a version of the game with only k units, each coalition controlling a single unit. This game has a pure equilibrium, since it is a simple congestion game. We will then repeatedly pick a coalition i that does not yet control $|P_i|$ units, and add a single unit to that coalition. For each addition, we inductively argue that the resulting game also has a pure equilibrium. In particular, we argue that if we start from a Nash equilibrium, and a single coalition is given an additional unit to place, natural game play will quickly converge to a stable solution. Thus, our result also suggests a natural method for efficiently computing pure equilibria in such a game.

Suppose we have a strategy $\mathcal{A}$ that is at equilibrium. Consider assigning an additional unit to some coalition i. Our first lemma characterizes coalition i's best response given that it has an extra unit to place.

Lemma 6.2.1 *Given that all other players assign their units as described by $\mathcal{A}$, a best* *response for coalition i is to add the new unit to one of the links without rearranging its existing allocation.*

Proof : Let A_i denote coalition i's strategy prior to the introduction of the additional unit. A coalition's strategy can be naturally viewed as a vector of length m, with each term indicating how many units are placed on the corresponding link. Consider the best response A_i' for coalition i (with the additional unit) that has minimum L_1-distance to A_i. It suffices to show that A_i' places at least as many units as A_i on any given link. Suppose otherwise. Then there is some link s on which i places fewer units under A_i' than under A_i. Since A_i' assigns more units in total, there must be a link t where A_i' places more units than A_i.

Consider starting from A_i' and moving one of i's units from t to s, creating A_i''. Since A_i'' has a smaller L_1-distance to A_i, it can't be a best response, and hence it must be beneficial to move that unit back from s to t. In particular, the marginal savings of removing a unit from s must strictly exceed the marginal cost of placing a unit on t. But now consider modifying A_i by moving a single unit from s to t. Since A_i has at most as many units on t as A_i'' does, and latencies are convex, the marginal cost of placing a unit on t under A_i is no greater than under A_i''. Likewise, the marginal savings of removing a unit from s it at least as large under A_i as it is under A_i''. Thus there is a positive net savings incurred by moving a unit from s to t under A_i. But this contradicts our assumption that we were at equilibrium before the arrival of the new unit. ■

We now know how coalition i will deal with an additional unit. Unfortunately, after placing that unit on link s, other coalitions may have an incentive to deviate. Observe that only those coalitions using link s could now have improving moves. Let j be a coalition using link s with an improving move (if none exist, we have an equilibrium), and imagine removing one unit controlled by that coalition from the game.

Lemma 6.2.2 *After removing one unit of coalition j from link s, the resulting strategy is at equilibrium.*

Proof : From the perspective of all coalitions other than i or j, the resulting strategy is indistinguishable from $\mathcal{A}$, and hence they do not have incentives to move. Furthermore, we claim that neither coalition i nor j will have an incentive to deviate. For coalition i this follows from an argument similar to that used in the proof of Lemma 6.2.1: the cost of his strategy A_i' only improved by removing one unit from s; if he had an improving move now, it can only be that he wants to move an extra unit to link s, but then this move would have been even more beneficial in the original strategy $\mathcal{A}$. For coalition j, all his remaining units have the same cost as they did in strategy $\mathcal{A}$. If he has an improving move, it would be to move a new unit to link s (where he has less players). However, we can argue that if coalition j had a such an improving move, then j would not have had an incentive to move in the first place. ■

Thus, ignoring the removed unit, all players are at equilibrium, and the number of units per link is the same as under $\mathcal{A}$. Hence the scenario is equivalent to the initial setting, only now, an extra unit must be assigned to coalition j, rather than i. Coalition j might place this unit on some link t, causing some other coalition using t to remove a unit, which must be then placed on another link, and so forth. We now consider a sequence of such moves and argue that it has bounded length.

Lemma 6.2.3 *The best response dynamic described above terminates.*

Proof : It suffices to show that if a coalition adds a unit to a link, it never removes a unit from that link. Suppose this is not the case for coalition i on link s. Over all possible sequences choose one of minimum length. Denote by $T_0, \ldots, T_f$ the time steps within this interval when coalition i deviates. Notice that i can not move units to or from s on any of the time steps $T_1, \ldots, T_{f-1}$.

Lemma 6.2.2 implies that during each of the time steps T_τ, coalition i removes a single unit from some link and moves it to another. We can decompose this process as follows: (a) Coalition i first removes the unit from a link, (b) the marginal cost of placing the removed unit on each link t is calculated (thus creating a list of numbers $\alpha_{\tau,t}$ for each time step T_τ), and (c) the extra unit is placed on a link t minimizing $\alpha_{\tau,t}$.

Consider the *smallest* number α_{τ^*,t^*} listed during this interval, where τ^* and t^* respectively denote the *earliest* time step and the link for which this number is observed. At time T_f the coalition moved a unit from link s to another link t, and thus $\alpha_{f,t} < \alpha_{f,s}$. Furthermore, $\alpha_{f,s} = \alpha_{0,s}$, since coalition i does not move any units to or from s during the observed interval. Since at time T_0, i places the unit on link s, all α values for this time are at least $\alpha_{0,s}$. This implies that the smallest α value listed over the entire interval does not occur during the first time step, i.e. $\tau^* \neq 0$. Thus, we can consider link t^* at time step $\tau^* - 1$.

By definition, $\alpha_{\tau^*-1,t^*} > \alpha_{\tau^*,t^*}$, and hence at time τ^*, coalition i must have removed a unit from link t^*. However, it must have been placed at a link other than t^* (otherwise, i would not have actually deviated). Hence there must be another link at this time step with a smaller α value, contradicting our assumption. ■

Combining the above lemmas yields the following result.

Theorem 6.2.4 *Congestion games with coalitions and convex latencies possess pure Nash equilibria.*

6.3 The Price of Collusion for Convex Latencies

In this section we will analyze the price of collusion for simple congestion games with convex latencies. As mentioned earlier, convex latencies arise naturally when modeling delay or response time as a function of congestion. We first prove our main result, that the price of collusion in such games is at most 2. We then present a sequence of games where the price of collusion tends toward 2, and hence this bound is tight.

The Upper Bound

Consider a game G for which all latencies are convex, and a set of coalitions defined by a partition P. For ease of presentation, we will assume that link latencies in G are distinct, i.e. $\ell_s(x) \neq \ell_t(y)$ for all s, t, x, y (this assumption can be dropped with only minor changes to the proof). A useful consequence of this assumption is that there is now a unique Nash equilibrium NE in the collusion-free game G.

For this section, we will use the following notation. Let CE be any Nash equilibrium in $G(P)$, the game with coalitions. Let n_s^i denote the number of units that coalition i puts on link s under CE. Let $n_s = \sum_i n_s^i$ denote the total number of units on link s under CE and let $n^i = \sum_s n_s^i$ denote the total number of units controlled by coalition i. Similarly, let n'_s denote the number of units on link s under NE. Finally, we denote the latency of link s under CE and NE as $\ell_s = \ell_s(n_s)$ and $\ell'_s = \ell_s(n'_s)$ respectively.

To make the calculations simpler in what follows, we start by showing that we can simplify the latency functions without loss of generality. We modify the latencies of the links without altering NE, CE or their social costs. For a link s, let $\alpha_s = \min(\ell_s, \ell'_s)$ and $\beta_s = \max(\ell_s, \ell'_s)$ denote the latencies of this link under NE and CE, and let $a_s = \min(n_s, n'_s)$ and $b_s = \max(n_s, n'_s)$ denote the corresponding loads. Define $\delta_s = \frac{\beta_s - \alpha_s}{b_s - a_s}$.

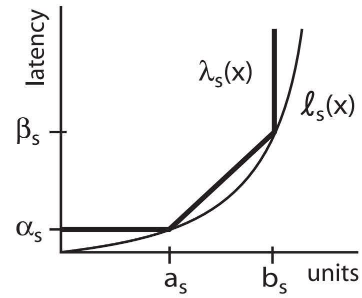

Figure 6.1: Simplifying the latency curves.

We now define a new latency function

$$\lambda_s(x) = \begin{cases} \alpha_s & \text{if } x < a_s \\ \alpha_s + (x - a_s)\delta_s & \text{if } a_s \leq x \leq b_s \\ \infty & \text{if } x > b_s \end{cases}$$

See the example shown in Figure 6.1. Notice that for the loads given by NE and CE on link s, the new latency function $\lambda_s(x)$ matches the original function $\ell_s(x)$. Furthermore, for any other load, $\lambda_s(x)$ is at least as large $\ell_s(x)$. Hence, both NE and CE are still Nash equilibria for their respective games G and $G(P)$. Moreover, the social costs of the two equilibria are unchanged and thus, from now on, we will assume without loss of generality that all latency functions $\ell_s(x)$ are of this simplified form.

We now proceed as follows. We partition the links into classes based on the loads they are given under NE and CE. We refer to $B = \{s \mid n_s = n'_s\}$ as the set of *balanced* links, and $U = \{s \mid n_s < n'_s\}$ as the *underloaded* links (i.e. those links for which CE has fewer units than NE). The links $L = \{s \mid n_s = n'_s + 1\}$ are called *lightly overloaded*, and finally the remaining links $H = \{s \mid n_s > n'_s + 1\}$ are called *heavily overloaded*. We call a link *overloaded* if it is in either L or H.

The main challenge will be to bound the cost paid by CE on the overloaded links, since on the remaining links, CE pays no more than NE. A key insight is that the heavily overloaded links are not problematic; in Lemma 6.3.3 we will argue that heavily overloaded links indicate a beneficial effect of collusion. Thus the heart of this proof lies in bounding the cost of the lightly overloaded links. We deal with these links first.

To begin, we group links from L and U into clusters as follows. Each cluster R contains some number d (which may vary for each cluster) of lightly overloaded links and a single underloaded link t such that $n_t \leq n'_t - d$. In other words, the underloaded edge has sufficient space to take one unit from all of the lightly overloaded links in R without becoming overloaded itself. We can choose any such grouping, so long as we ensure that each lightly overloaded link appears in some cluster. We will use NE_R and CE_R to denote NE and CE restricted to the links in R.

For each cluster R, we construct an intermediate allocation $\mathcal{I}_R$ by starting with CE_R and moving a single unit from each lightly overloaded link to the underloaded one. Thus, $\mathcal{I}_R$ is identical to NE on the overloaded links, and puts exactly d units more than CE on the underloaded link t.

Let us assume without loss of generality that the cluster R consists of the first $d+1$ links, with the first d of these being the lightly overloaded links, and the $(d+1)$-st link being the underloaded one. We will use δ to denote δ_{d+1}, and κ to denote the number of coalitions participating in any of the first d links. The following lemma will be crucial in bounding the cost of the CE_R.

Lemma 6.3.1 *For all coalitions i participating on any of the links $1, \ldots, d$, $n^i_{d+1} \geq d$. In other words, under CE, any coalition that uses an overloaded link in R must have at least d units on the underloaded link.*

Proof : For the sake of contradiction, assume that coalition i has at least one unit on some overloaded link $s \in \{1, \ldots, d\}$, and has at most $d-1$ units on the last link. As

CE was a stable state for this coalition, it has no incentive to move a unit from link s to link $(d+1)$. The saving it gets from taking the unit off of s is at least ℓ_s, the current latency at of that link. The extra cost it incurs by adding that unit to link $(d+1)$ is at most $(d-1)\delta + (\ell_{d+1} + \delta) = d \cdot \delta + \ell_{d+1}$ (the first part accounts for the extra cost that all of its units on the last link will incur, and the second part accounts for the latency of the new unit). Hence $\ell_s \leq d \cdot \delta + \ell_{d+1}$.

On the other hand, $\ell_s > d \cdot \delta + \ell_{d+1}$, since otherwise a player in the collusion-free Nash equilibrium NE (which has at least d more units on link $(d+1)$ than CE) would have an incentive to move from link $(d+1)$ to link s and incur a lower cost[1]. Thus we have a contradiction. ■

We can now bound the cost of CE_R in terms of $\mathcal{I}_R$.

Lemma 6.3.2 *For any cluster C, $c(CE_R) \leq 2c(\mathcal{I}_R)$*

Proof : It is sufficient to prove that the cost difference $\Delta = c(CE_R) - c(\mathcal{I}_R)$ is at most $c(\mathcal{I}_R)$. To do this, we need a lower bound on $c(\mathcal{I}_R)$, and an upper bound on Δ. Recall that $\mathcal{I}_R$ puts exactly d units more on link $(d+1)$ than CE. These extra units increase the cost paid by the original n_{d+1} units on this link by at least δ (as $d \geq 1$), so the original n_{d+1} units on link $(d+1)$ pay at least $n_{d+1}(\ell_{d+1} + \delta)$. Counting only the cost of these n_{d+1} units we get $c(\mathcal{I}_R) \geq n_{d+1}(\ell_{d+1} + \delta)$. Lemma 6.3.1 implies that $n_{d+1} \geq \kappa \cdot d$, where κ is the number of coalitions participating in the first d links. So we get the lower bound we will use:

$$c(\mathcal{I}_R) \geq n_{d+1}(\ell_{d+1} + \delta) \geq \kappa \cdot d \cdot (\ell_{d+1} + \delta). \tag{6.1}$$

[1]This is the only place we use the fact that latencies are distinct. This assumption can be removed by altering the proof slightly.

Now consider the difference $\Delta = c(CE_R) - c(\mathcal{I}_R)$. First we express Δ in terms of n_s, ℓ_s and δ_s. Then we consider each of the κ coalitions participating on any of the first d links. None of these coalitions has an incentive to move one unit from an earlier link to link $(d+1)$. We use this to give an inequality regarding the delays on the links. Summing over all coalitions will allow us to bound the difference Δ.

We can express the difference Δ exactly in terms of the quantities n_s, ℓ_s and δ_s as follows. The cost of CE_R on any link $s \in \{1 \ldots d\}$ exceeds the costs of $\mathcal{I}_R$ on that link by exactly $(\ell_s - \delta_s) + n_s \cdot \delta_s$, since the two configurations differ by only a single unit. On link $(d+1)$, $\mathcal{I}_R$ puts d units more than CE_R, so it overpays by $(n_{d+1} + d)(\ell_{d+1} + d \cdot \delta) - n_{d+1} \cdot \ell_{d+1}$. Hence,

$$\Delta = \sum_{s \in [1 \ldots d]} [(\ell_s - \delta_s) + n_s \cdot \delta_s] - [(n_{d+1} + d)(\ell_{d+1} + d \cdot \delta) - n_{d+1} \cdot \ell_{d+1}]. \qquad (6.2)$$

Since CE is a coalition equilibrium, no coalition i has an incentive to move a unit from any link $s \in \{1, \ldots, d\}$ to link $(d+1)$. Hence for any coalition i participating on link s, we have $(\ell_s - \delta_s) + n_s^i \cdot \delta_s \leq (\ell_{d+1} + \delta) + n_{d+1}^i \cdot \delta$. Summing this inequality over all coalitions on link s yields

$$k_s(\ell_s - \delta_s) + n_s \cdot \delta_s \leq k_s(\ell_{d+1} + \delta) + \sum_{i \ on \ s} n_{d+1}^i \cdot \delta \leq \kappa(\ell_{d+1} + \delta) + n_{d+1} \cdot \delta, \qquad (6.3)$$

where k_s denotes the number of coalitions participating on link s and recall that κ denotes the number of coalitions participating on the first d links. Since $k_s \geq 1$, we can apply inequality 6.3 to equation 6.2 yielding

$$
\begin{aligned}
\Delta &\leq \sum_{s\in[1\ldots d]} [\kappa(\ell_{d+1}+\delta)+n_{d+1}\cdot\delta] - [(n_{d+1}+d)(\ell_{d+1}+d\cdot\delta) - n_{d+1}\cdot\ell_{d+1}] \\
&= d\cdot\kappa\cdot\ell_{d+1} + d\cdot\kappa\cdot\delta + d\cdot n_{d+1}\cdot\delta - n_{d+1}\cdot d\cdot\delta - d\cdot\ell_{d+1} - d^2\cdot\delta \\
&\leq d\cdot\kappa(\ell_{d+1}+\delta). \qquad (6.4)
\end{aligned}
$$

Note that our upper bound on Δ matches our lower bound for $c(\mathcal{I}_R)$, completing the proof. ■

We can now define an intermediate allocation $\mathcal{I}$ for all links by using the intermediate configurations $\mathcal{I}_R$ for each cluster R and using the allocation given by CE for all remaining links. Lemma 6.3.2 implies that $c(CE) \leq 2\cdot c(\mathcal{I})$. It remains to show that the cost of $\mathcal{I}$ does not exceed that of NE.

Lemma 6.3.3 *Let $s \in H$ be a heavily overloaded link, let $t \in U$ be an underloaded link, and let $\mathcal{A}$ be the strategy that results from starting at CE and moving a single unit from s to t. Then $c(\mathcal{A}) \geq c(CE)$.*

Proof : Consider an overloaded link s. Suppose there exists an underloaded link t, such that moving a single unit from s to t improves the cost of CE. It will suffice to show that $n_s = n'_s + 1$. In other words, we will argue that s was in fact only lightly overloaded.

Under CE, the total cost of the units on links s and t is $n_s\ell_s + n_t\ell_t$. For $\mathcal{A}$, the total cost of these links is $(n_s-1)(\ell_s-\delta_s)+(n_t+1)(\ell_t+\delta_t)$. These two solutions have the same cost for all other links. Hence, by our assumption, we have the following inequality:

$$n_s\ell_s + n_t\ell_t > (n_s-1)(\ell_s-\delta_s)+(n_t+1)(\ell_t+\delta_t). \qquad (6.5)$$

On the other hand, we know that CE is an equilibrium for all coalitions, and thus no coalition has incentive to move one unit from s to t. In particular, for any coalition i who participates on s we must have

$$n_s^i \ell_s + n_t^i \ell_t \leq (n_s^i - 1)(\ell_s - \delta_s) + (n_t^i + 1)(\ell_t + \delta_t).$$

Summing this inequality over all coalitions who participate on s, yields

$$\sum_i n_s^i \ell_s + \sum_i n_t^i \ell_t \leq \sum_i (n_s^i - 1)(\ell_s - \delta_s) + \sum_i (n_t^i + 1)(\ell_t + \delta_t). \tag{6.6}$$

where the summation for i is running only over all coalitions that have a unit on link s. Now adding the trivial inequality $n_t^i \ell_t \leq n_t^i(\ell_t + \delta_t)$ for coalitions that participate in link t but not in link s, we get that the left hand side is equal to $n_s \ell_s + n_t \ell_t$, matching that of inequality (6.5). The right side is $(n_s - 1)(\ell_s - \delta_s) + (n_t + 1)(\ell_t + \delta_t) - (k_s - 1)(\ell_s - \delta_s) + (k_s - 1)(\ell_t + \delta_t)$, where k_s denotes the number of coalitions participating on link s. Notice that this latter quantity matches the right hand side of inequality (6.5) with the addition of the term $(k_s - 1)[(\ell_t + \delta_t) - (\ell_s - \delta_s)]$. Therefore for the inequalities 6.5 and 6.6 to hold simultaneously, it must be the case that $(k_s - 1)[(\ell_t + \delta_t) - (\ell_s - \delta_s)] > 0$. Thus $(\ell_t + \delta_t) > (\ell_s - \delta_s)$. Since t was underloaded, $\ell_t + \delta_t \leq \ell'_t$. But we also know that $\ell'_t \leq \ell'_s + \delta_s$, since NE is a stable configuration without coalitions and thus no unit should benefit by moving from t to s. Combining these last three inequalities yields

$$\ell'_s + \delta_s > \ell_s - \delta_s. \tag{6.7}$$

Recall that ℓ_s and ℓ'_s were the latencies of this link under CE and NE respectively, and adding (subtracting) δ_s gives the latency with one more (less) unit. Therefore, inequality (6.7) implies that the difference between the number of units that CE and NE put on link j can not be more than one. ■

We now prove that the intermediate solution is no more costly than the collusion-free Nash equilibrium.

Lemma 6.3.4 $c(\mathcal{I}) \leq c(NE)$.

Proof : Consider converting allocation $\mathcal{I}$ to NE by moving a unit at a time from an overloaded link to an underloaded one. Recall that the construction of $\mathcal{I}$ ensured that the only overloaded links are heavily overloaded. Furthermore, Lemma 6.3.3 implies that moving a unit away from a heavily overloaded link H to an underloaded link in U does not decrease social cost. Since in $\mathcal{I}$ the underloaded links have no fewer units than in CE, this property still holds. The convexity of latencies implies that if moving one unit between two links increases the social cost, then moving subsequent units will further increase the social cost. Hence in each step of the conversion from $\mathcal{I}$ to NE the social cost only increases, and thus $c(\mathcal{I}) \leq c(NE)$. ■

Taken together, Lemmas 6.3.2 and 6.3.4 prove our main result.

Theorem 6.3.5 *The price of collusion for congestion games with convex latencies is at most 2.*

We conclude with an example demonstrating that this bound is tight.

Example : Consider an instance of the game with two links and $2k$ units. Defined latencies as follows.

$$\ell_1(x) = \begin{cases} 0 & x \leq k-1 \\ 2 & x = k \\ \infty & x \geq k+1 \end{cases} \qquad \ell_2(x) = \begin{cases} 0 & x \leq k \\ 1 & x = k+1 \\ \infty & x \geq k+2 \end{cases}$$

Nash equilibrium without coalitions places $k-1$ units on the first link and $k+1$ on the second, for a total cost of $k+1$. Now form k coalitions of size 2. If each coalition places one unit on both links, the resulting strategy is at equilibrium; moving a unit from the first link to the second does not yield an improvement in cost. The total cost of this solution is $2k$. Thus the price of collusion for this game is $\frac{2k}{k+1}$. ■

6.4 The Price of Collusion for Concave Latencies

In this section we consider simple congestion games for which latencies are concave and increasing. We give a simple argument showing that the price of collusion for these games is bounded by 2 if we assume that pure equilibria exist after coalitions are formed. Since we do not know that this is always the case, we also provide a weaker bound for the price of collusion under mixed strategies, which are guaranteed to exist. Lastly, we give an example of a game with concave latencies for which the price of collusion is non-trivial (strictly greater than 1), although unlike our convex example, this does not match our upper bound.

First, consider a collusion-free equilibrium NE. We start with a simple claim.

Claim 6.4.1 *At NE the largest latency L and the smallest latency ℓ differ by at most a factor of two.*

Proof : Assume that $2\ell < L$ and consider the link with latency ℓ. Due to concavity, adding one more unit to that link can not increase its latency beyond $2 \cdot \ell < L$. Hence a unit on the link with latency L has incentive to switch, contradicting the assumption of NE. ■

This allows us to prove the following result.

Theorem 6.4.2 *For any pure coalition equilibrium CE, $c(CE) \leq 2 \cdot c(NE)$.*

Proof : Let ℓ and L denote respectively the smallest and the largest latencies in NE. Claim 6.4.1 implies that $L \leq 2 \cdot \ell$. Obviously, $c(NE) \geq n \cdot \ell$.

We call a link s *underloaded* if its load at CE is less than its load at NE, i.e. $n_s < n'_s$. Consider a particular coalition i with n^i units and imagine that coalition extracts its units from CE, possibly creating additional underloaded links. Now consider what happens when i places its units on underloaded links, such that the resulting load of those links does not exceed their loads at NE. Such a placement is clearly possible, since the total number of units of NE and CE are identical.

The cost of such a move is at most $n^i \cdot L$. Since CE is a coalition equilibrium, this is an upper bound on what coalition i pays in CE. Summing this bound over all coalitions yields that the total cost $c(CE) \leq \sum_i n^i \cdot L = n \cdot L \leq 2 \cdot n \cdot \ell \leq 2 \cdot c(NE)$, completing the proof. ■

We now present a bound for mixed equilibria, which are known to always exist. In a mixed equilibrium, players select distributions over pure strategies so as to minimize the expected cost they incur given all other players' distributions. Note that for any pure strategy A_i, player i must evaluate the cost of A_i by measuring the expected cost given that other players act according to their distributions, and that i *deterministically* plays A_i. A randomized strategy for player i is a distribution over deterministic strategies, and hence if a player plays a randomized strategy, the expected load experienced by the player may exceed the expected load of the machines he is randomizing between. Because of this we lose a factor of 2 in our bound.

Theorem 6.4.3 *For any coalition equilibrium CE, $\mathrm{E}\left[c(CE)\right] \leq 4 \cdot c(NE)$.*

Proof Sketch: Observe that, as in the previous result, we are comparing CE to a pure collusion-free equilibrium. We now say that a link s is underloaded if the expected load on s under CE is less than the deterministic load given by NE. A mixed strategy is at equilibrium if no player i has any deterministic strategy that has better payoff than the mixed strategy he is playing.

The proof follows the same outline as in the pure case. For any coalition i, we consider removing i's units, and placing them deterministically on underloaded links. Since links may be fractionally underloaded, this may yield an allocation where the expected load on some links exceeds the load under NE. However, we can reassign coalition i's units so that the expected load X on any link s exceeds the load given by NE by at most 1 unit. Since latencies are concave, $\mathrm{E}\big[\ell_s(X)\big] \leq \ell_s(\mathrm{E}\big[X\big])$. Thus the expected per unit cost for coalition i is at most the latency of a link in NE overloaded by a unit. By Claim 6.4.1 the maximum expected latency under NE is at most $L \leq 2\ell$. By concavity, the latency of any link increases by at most a factor of 2 when overloaded by 1, so the average latency for any player is at most $2L \leq 4\ell$. ■

We conclude by giving an example where the price of collusion with concave latencies is 8/7.

Example : Consider an instance of the game with 2 links and 4 players. The latencies are defined as follows: The first link has latency $\ell_1(1) = 1/2$ and $\ell_1(x) = 1$ for all $x > 1$, while the second has latency $\ell_2(x) = 1 + (x - 4)\epsilon$. NE will place a single unit on the first link and three units on the second, for a total cost of $\frac{1}{2} + 3(1 - \epsilon) \sim 3.5$. If we now form two coalitions of size 2, we have a coalition equilibrium in which both coalitions put a single unit on each link. The resulting outcome has a cost of $2 + 2(1 - 2\epsilon) \sim 4$. Hence the price of collusion for this game is $8/7$. ■

Chapter 7

Conclusion

In this chapter we discuss some open problems concerning the games discussed in this thesis. While clearly not an exhaustive list, the following are some of the most natural and intersting questions to arise from our work.

The Connection Games

In the connection game with arbitrary cost sharing (chapter 3), we show that there is always a 3-approximate equilibrium that builds an optimal network, and that exact equilibria (1-approximate) might not exist at all. More generally, we can think of our results as having a bicriteria flavor: define a solution to be (α, β)-approximate when it is an α-approximation to the optimal centrally designed solution and a β-approximate equilibrium. In these terms, we have shown that $(1, 3)$-approximate equilibria always exist, and that $(\alpha, 1)$-approximate equilibria need not, regardless of how large α is. It would be interesting to map out the space of α and β parameters for which such solutions exist.

In our definition of the connection game with fair cost sharing, player strategies consist of simply picking paths. Thus, it would be natural to define the connection game with arbitrary cost sharing analogously; players pick paths, along with payments towards those paths. However, we make no such restriction that players do contribute to a particular path. In the case of exact (non-approximate) equilibria, these definitions are equilivalent, since obviously no player would ever willingly contribute to a edge that he does not use on his connecting path. But in the case of approximate equilibria, this distinction becomes more important. In particular, our approximation results take advantage of this definition by having a small portion of the network paid for by players without regard to whether those players actually use the edge or not. Even if we accept the value of considering approximate equilibria, this scenario seeks awkward. It is not clear how to alter the current game definition without invalidating some of our results. However, if this could be done, it would not only allow for a more natural game description, but also imply an improved lower bound; under this definition, a Nash equilibrium that buys the optimal network may need to be at least 2-approximate. The example for this is simply a cycle of length $2k$, where each player has terminals at a distinct pair of opposite nodes.

One very intriguing question to come out of our work on the connection game with fair cost sharing is the role of directed edges. The example we use to show that the price of stability is $O(\log(k))$ requires the graph be directed. If we make this graph undirected, the price of stability immediately drops to 1. As shown in chapter 4, there are examples of undirected games for which the price of stability is larger, but these do not seem to generalize in a natural way, and all examples we have seen have a small price of stability. It seems unlikely that our $O(\log(k))$ bound is tight for undirected graphs, and possibly even a bound as small as 2 is attainable.

Best Response Dynamics

At the end of chapter 4, we provide an example of best response dynamics that runs in exponential time, along with a proof that for two players, gameplay always converges in time that is linear in the number of nodes in the network. While this covers two extreme cases, it leaves open a large middle-ground of problems. Notably, it remains unknown whether three players will converge to a stable solution in polytime, and whether a suitable ordering of best response dynamics might always lead to polynomial convergence. We tentatively conjecture an affirmative answer to both of these questions. In our example of exponential best response dynamics, even just ordering players in a round-robin fashion yields efficient convergence.

Beyond the fact that equilibria do not always exist, this game might be too complex a setting for an initial study of the convergence of best response dynamics. A congestion game similar to those studied in chapter 6 might provide a better setting to study how the order of players effects the time required for players to reach a stable state, as existence of equilibria and the eventual convergence are both guaranteed. Of particular interest would be a game for which exponential runs exist, but for which a clever (or even not so clever) ordering of players suffices to ensure fast convergence.

The Network Pricing Game

The most obvious unresolved question from chapter 5 is to determine the actual price of anarchy for our pricing game. Currently, our lower bound is 1.5, while our upper bound exceeds 4. There is clearly a significant amount of slack in the analysis of the latter, and possibly by tightening this, a better bound can be found. It is notable that our lower bound involves a single player, i.e. a monopolist. If it happens to be the case that the worst case upper bound can always be achieved with a single player, this would not only be a surprising fact, but presumably provide a cleaner and improved upper bound as well.

There are a number of natural generalizations of this pricing game that might prove interesting. One would be to model heterogenous users. While the disutility curve we consider in our game does model the fact that users differ in the quality of service they demand, they all measure quality of service in exactly the same way. In particular, they all value time (latency) and money (cost) equivalently, since they experience disutility as the unweighted sum of these two values. A more realistic and interesting model would allow each network user to assign a weight to the time (or money) component of this sum. Thus we could model the fact that some users are willing to pay a lot for a fast connection, while others are willing to endure a slow connection if it is cheap enough. The simplest version of this game would have two classes of users, each with their own trade-off parameter. In general, we could have many classes of users, or even a continuous distribution over the trade-off parameter. However, the variants we have considered yield games that are either not well defined, or fail to have pure equilibria.

A more direct extension would be to consider pricing in more general networks. One problem here is that if we are interested in modeling multi-commodity demands, then we can effectively encode the example with no pure equilibria from section 5.4 while still using linear latencies. However, it is still possible that in the case of a single commondity, or even for a single source game, pure equilibria do exist. In this case, we might hope for a good upper bound on the price of anarchy. One concern is that any such bound would be at least linear in the length of the longest path. A well known example of this game involves a path of k links, no latency, and a linear decreasing disutility. Even here the the price of anarchy is $O(k)$. However, such a result would still be interesting, as many networks do have a bounded depth.

Pure Equilibria

In chapter 5, the specific game we study was one of the few such network pricing games we initially examined for which pure equilibria always exist (and even then, only with the rather strong assumptions of linear latencies and parallel links). As discussed, we prefer to consider pure equilibria, as mixed equilibria are not necessarily realistic strategies for players in certain large-scale network games. Furthermore, analysis of mixed strategies tends to be significantly more difficult. But in general, relatively few classes of games beyond potential games and some variants are known to have pure equilibria.

In light of this, one approach would be to focus on approximate Nash equilibria. The notion of approximate equilibria used in this thesis is multiplicative; players in an α-approximate equilibrium can benefit by at most a factor of α in changing their strategies. But we could also consider other notions, such as additive approximate equilibria. Such equilibria would correspond to game with fixed costs for changing stratgies. Presumably such a factor would need to depend on the utilities of players, as games are effectively scale-free; multiplying the utilities of each player by a constant does not change the game.

Alternatively, we could consider other notions of equilibria, such as the sink-equilibria proposed by Mirrokni and Vetta [50]. While not stable states like Nash equilibria, these are instead collections of states from which best response dynamics do not escape. For the class of games considered by the authors, best response dynamics quickly converge to such a sink equilibria, and the quality of all of the states in this sink-equilibria are proved to be within a constant factor of a central optimum. Applying this notion of equilibrium to other settings would allow us to still consider only pure strategies, even when pure Nash equilibria might not exist at all.

Static Coalitions in Congestion Games

Two of the strongest restrictions on the class of congestion games we consider in chapter 6 are that of symmetry and our requirement that strategies must correspond to singleton sets. It does not seem possible to drop these restrictions entirely; we show that the price of collusion can increase to $O(k)$ in an asymmetric game, and with multi-element strategies we lose many structural properties of pure equilibria. However, perhaps by limiting the degree of asymmertry, or by restricting the types of strategy sets available to players, a strong bound can still be achieved.

We turned to mixed equilibria as our proof that pure equilibria exist only applies to the case of convex latencies. However, we know of no example that indicates that pure equilibria do not exist with concave latencies after the formation of coalitions. In fact, we do not have such an example for general increasing latencies. In the general case, we can not hope for a small bound on the price of collusion, but either a positive or a negative answer regarding the existence of pure equilibria would most likely be illuminating.

Lastly, the extent to which our proof techniques for the price of collusion with convex latencies extend to deal with mixed equilibria remains open.

Dynamic Coalitions

Chapter 6 introduces our model of static coalitions. In this framework, player coalitions are dictated externally; players are simply told who they will be cooperating with, and have no choice but to do so. An even more interesting line of questioning asks what happens when even this choice of whom to collude with is a part of players' strategy spaces. In such a dynamic coalition, any player would have the option of deviating to not only another stragey, but also another coalition, if doing so was beneficial. As in chapter 6, we are not interested in preventing collusion, but instead would like a model in which coalitions may form and be part of a final stable outcome. Clearly such a model is appealing, but so far we have been unable to develop a reasonable formulation of dynamic coaltions.

For a simple indication in the difficulty inherent in modeling dynamic coalitions, consider the following line of reasoning. We would like any such model to capture the scenario in which two players choose to collude and decide that the first will pay the second to act against his own best interests. In other words, the second player makes a non-optimal move, and as a result loses some utility, but meanwhile the first player recieves an even more significant gain in utility, some of which he uses as a side payment to induce the second player to agree to this deal.

The problem arises if now the first player has the option of switching strategies. For example, the first player could simply switch to the strategy wherein he makes the same action as before, but now doesn't collude with anyone, and in particular, doesn't pay the second player anything. Obviously if we allow this, then there will be no stable collusion with side payments. If we explicitly forbid this type of defection, we might still have the situation where the first player deviates to another action that is effectively identical to his current action, and thus break out of his current coalition. It seems that the only natural way to stop this behavior is to allow player two maintain a threatened move in the case that player one defects. Beyond the complexity of such a description when coalitions are large, we have the problem that now the first player must decide whether or not to defect by anticipating what the second player will do. This sort of decision making is antithetical to the very notion of Nash equilibria.

One approach to such a problem is to simply not allow side payments. While this is certainly worth consideration, it doesn't really capture the dynamics we are interested in. A more appealing approach is to study a more limited form of dynamic coalitions, such as in the work of Johari, Mannor, and Tsitsiklis [39]. In this paper, the authors consider a game in which players control nodes and attempt to form a network. Players can form pair-wise contracts with other nodes to form a link between them. Their utility depends on the side-payments they agree upon, as well as the resulting network topology. Thus the model allows for a limited form of two-player dynamic "coalitions," but avoids the problems discussed above by requiring that each coalition effectively makes a binary decision; to either build a link or not. Thus the net effect of breaking a coalition can be directly and simply anticipated by a player. Generalizing such a model seems a promising first step towards understanding the effects of coalitions on games.

BIBLIOGRAPHY

[1] D. Acemoglu and A. Ozdaglar. Flow control, routing, and performance from service provider viewpoint. Unpublished manuscript, MIT 2004.

[2] D. Acemoglu and A. Ozdaglar. Competition and efficiency in congested markets. Unpublished manuscript, MIT 2005.

[3] P. Afèche. Incentive-compatible revenue management in queuing systems: optimal strategic idleness and other delaying tactics. Unpublished manuscript, Northwestern University 2004.

[4] A. Agrawal, P. Klein, and R. Ravi. When trees collide: An approximation algorithm for the generalized Steiner problem on networks. *SIAM Journal on Computing*, 24(3) 445-56 (1995).

[5] A. Akella, R. Karp, C. Papadimitrou, S. Seshan, and S. Shenker. Selfish Behavior and Stability of the Internet: A Game-Theoretic Analysis of TCP. In *Proceedings of the ACM SIGCOMM 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM)*, 2002.

[6] G. Allon and A. Federgruen. Competition in service industries. Unpublished manuscript, Columbia University 2004.

[7] E. Anshelevich, A. Dasgupta, J. Kleinberg, É. Tardos, T. Wexler, and T. Roughgarden, The price of stability for network design with fair cost allocation. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS)* 2004: 295-304.

[8] E. Anshelevich, A. Dasgupta, É. Tardos, and T. Wexler. Near-optimal network design with selfish agents. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)* 2003: 511-20.

[9] 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)* 2005: 57-66.

[10] V. Bala and S. Goyal. A Non-Cooperative Model of Network Formation. *Econometrica* 68(2000), 1181-229.

[11] M. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the Economics of Transportation.* Yale University Press, 1956.

[12] S. Catoni and S. Pallottino. Traffic equilibrium paradoxes. *Transportation Science* 25 (1991): 240-4.

[13] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha. Approximation Algorithms for Directed Steiner Problems. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1998.

[14] H. Chen and T. Roughgarden. Network design with weighted players. Unpublished manuscript, Stanford University 2005.

[15] 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)* 2005: 67-73.

[16] G. Christodoulou and E. Koutsoupias. On the price of anarchy and stability of correlated equilibria of linear congestion games. Unpublished manuscript, University of Athens 2005.

[17] J.R. Correa, A. Schulz, and N. Stier Moses. Selfish routing in capacitated networks. *Mathematics of Operations Research* 29 (2004): 961-76.

[18] 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)* 2003: 521-30.

[19] R. Cole, Y. Dodis, and T. Roughgarden. How much can taxes help selfish routing. In *Proceedings of the 4th ACM Conference on Electronic Commerce (EC)* 2003: 98-107.

[20] A. Czumaj, P. Krysta, and B. Vöcking. Selfish traffic allocation for server farms. In *Proceedings of the 34th Annual ACM Symposium on the Theory of Computing (STOC)* 2002: 287-96.

[21] A. Czumaj and B. Vöcking. Tight bounds for worst-case equilibria. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* 2002: 413-20.

[22] D. Dutta A. Goel, and J. Heidemann. Oblivious AQM and Nash Equilibrium. In *Proceedings of the IEEE Infocom* 2003.

[23] A. Fabrikant, A. Luthra, E. Maneva, S. Papadimitriou, and S. Shenker. On a network creation game. In *Proceedings of the 22nd Annual ACM Symposium on Principles of Distributed Computing (PODC)* 2003: 247-51.

[24] A. Fabrikant, C. Papadimitriou, K. Talwar. The complexity of pure Nash equilibria. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC),* 2004.

[25] U. Feige. A treshold of $\lg n$ for approximating set-cover. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC)* 1996: 314-18.

[26] J. Feigenbaum, C. H. Papadimitriou, S. Shenker. Sharing the cost of multicast transmissions. *Journal of Computer and System Sciences* 63 (1) pages 21-41, 2001.

[27] 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)* 2004.

[28] D. Fudenberg and J. Tirole. *Game Theory.* MIT Press, 1991.

[29] M. Goemans and D. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal on Computing*, 24:296-317, 1995.

[30] A. Goldberg and J. Hartline. Collusion-resistant mechanisms for single-parameter agents. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* 2005: 620-9.

[31] G. Hardin. The tragedy of the commons. *Science*, 162:1243-8, 1968.

[32] A. Hayrapetyan, É. Tardos, and T. Wexler. A network pricing game for selfish traffic. In *Proceedings of the 24th Annual ACM Symposium on Principles of Distributed Computing (PODC)* 2005.

[33] A. Hayrapetyan, É. Tardos, and T. Wexler. A network pricing game for selfish traffic. Unpublished manuscript, Cornell University 2005.

[34] L. He and J. Walrand. Pricing internet services with multiple providers. In *Proceedings of the 41st Annual Allerton Conference on Communication and Control* 2003.

[35] H. Heller and S. Sarangi. Nash Networks with Heterogeneous Agents. Working Paper Series 2001, E-2001-1, Virginia Tech.

[36] Shai Herzog, Scott Shenker, Deborah Estrin. Sharing the ”Cost” of Multicast Trees: An Axiomatic Analysis. *IEEE/ACM Transactions on Networking*, Dec. 1997.

[37] K. Jain and V. Vazirani. Applications of approximation algorithms to cooperative games. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)* 2001: 364-72.

[38] K. Jain and V. Vazirani. Equitable cost allocations via primal-dual-type algorithms. In *Proceedings in the 34th Annual ACM Symposium on Theory of Computing (STOC)* 2002: 313-21.

[39] R. Johari, S. Mannor, and J. N. Tsitsiklis. A contract-based model for directed network formation. Unpublished manuscript, 2003.

[40] R. Johari and J. Tsitsiklis. Efficiency loss in a network resource allocation game. *Mathematics of Operations Research*, to appear.

[41] G. Karakostas and S. Kolliopoulos. Edge pricing of multicommodity networks for heterogeneous selfish users. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS)* 2004.

[42] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, volume 8 (1997), 33-7.

[43] K. Kent and D. Skorin-Kapov. Population monoton cost allocation on mst's. In *Operations Research Proceedings* KOI 1996: 43-8.

[44] A. Kothari, S. Suri, C. D. Tóth, and Y. Zhou. Congestion games, load balancing, and price of anarchy. In *Workshop on Combinatorial and Algorithmic Aspects of Networking* 2004.

[45] E. Koutsoupias and C. Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science* 1999: 404-13.

[46] L. Libman and A. Orda. Atomic Resource Sharing in Noncooperative Networks. *Telecommunication Systems*, 17:4, 385-409, 2001.

[47] I. Lee and R. Mason. Market structure in congestible markets. *European Economic Review*, 45 (4-6) 809-18, 2001.

[48] M. Mavronicolas and P. Spirakis. The price of selfish routing. In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing (STOC)* 2001: 510-9.

[49] I. Milchtaich. Congestion games with player-specific payoff functions. *Games and Economic Behavior* 13 (1996): 111-24.

[50] V. S. Mirrokni and A. Vetta. Convergence issues in competitive games. In *Proceedings of the 7th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)* 2004: 183-94.

[51] D. Monderer, D. Samet. Variations of the shapley value. *Handbook of Game Theory Vol. III* (Ed. R.J. Aumann and S. Hart), Elsevier Science, 2002.

[52] D. Monderer, L. Shapley. Potential games. *Games and Economic Behavior* 14 (1996): 124-43.

[53] H. Moulin, S. Shenker. Strategyproof sharing of submodular costs: budget balance versus efficiency. *Economic Theory* 18 (1996): 511-33.

[54] C. Papadimitriou. Algorithms, games, and the Internet. In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing (STOC)* 2001: 749-53.

[55] M. Pál and É. Tardos. Group strategyproof mechanisms via primal-dual algorithms. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS)* 2003: 584-93.

[56] A. C. Pigou. *The Economics of Welfare.* Macmillan and Co., 1920.

[57] G. Robins and A. Zelikovsky Improved Steiner Tree Approximation in Graphs. In *Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* 2000: 770-9.

[58] R. W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory* 2 (1973): 65-7.

[59] R. W. Rosenthal. The network equilibrium problem in integers. *Networks,* 3:53-59, 1973.

[60] T. Roughgarden. Stackelberg scheduling strategies. In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing (STOC)* 2001: 104-13.

[61] T. Roughgarden. The price of anarchy is independent of the network topology. In *Proceedings of the 34th Annual ACM Symposium on the Theory of Computing (STOC)* 2002: 428-37.

[62] T. Roughgarden. Selfish routing with atomic players. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* 2005: 1184-5.

[63] T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the ACM* 49(2) (March 2004): 236-59.

[64] A. Schulz and N. Stier Moses. On the Performance of User Equilibria in Traffic Networks. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003.

[65] S. Shenker. Making Greed Work in Networks: A Game-Theoretic Analysis of Switch Service Disciplines. *IEEE/ACM Transactions on Networking*, 819-31, 1995.

[66] S. Suri, C. D. Tóth, and Y. Zhou. Selfish load balancing and atomic congestion games. In *Proceedings of the 16th Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)* 2004: 188-95.

[67] A. Vetta. Nash equilibria in competitive societies with applications to facility location, traffic routing and auctions. In *Proceedings of the 43th Annual IEEE Symposium on Foundations of Computer Science (FOCS)* 2002.