How Much Can Taxes Help Selfish Routing?*

Richard Cole† Yevgeniy Dodis‡ Tim Roughgarden§

July 28, 2005

Abstract

We study economic incentives for influencing selfish behavior in networks. We consider a model of selfish routing in which the latency experienced by network traffic on an edge of the network is a function of the edge congestion, and network users are assumed to selfishly route traffic on minimum-latency paths. The quality of a routing of traffic is historically measured by the sum of all travel times, also called the *total latency*.

It is well known that the outcome of selfish routing (a *flow at Nash equilibrium*) does not minimize the total latency, and that *marginal cost pricing*—charging each network user for the congestion effects caused by its presence—eliminates the inefficiency of selfish routing. However, the principle of marginal cost pricing assumes that taxes cause no disutility to network users; this is appropriate only when collected taxes can be feasibly returned (directly or indirectly) to the users. If this assumption does not hold and we wish to minimize the total user disutility (latency plus taxes paid)—the total *cost*—how should we price the network edges? Intuition may suggest that taxes can never improve the cost of a Nash equilibrium, but the famous *Braess's Paradox* shows this intuition to be incorrect.

We consider strategies for pricing network edges to reduce the cost of a Nash equilibrium. Since levying a sufficiently large tax on an edge effectively removes it from the network, our study generalizes previous work on designing networks for selfish users [36]. In this paper, we prove the following results.

- In a large class of networks—including all networks with linear latency functions—marginal cost taxes do not improve the cost of a Nash equilibrium.

*A preliminary version of this paper appeared in the Proceedings of the 4th ACM Conference on Electronic Commerce, June 2003.

†Department of Computer Science, New York University, 251 Mercer Street, New York, NY 10012. Supported in part by NSF grant CCR0105678. Email: cole@cs.nyu.edu.

‡Department of Computer Science, New York University, 251 Mercer Street, New York, NY 10012. Supported in part by an NSF CAREER Award. Email: dodis@cs.nyu.edu.

§Department of Computer Science, Stanford University, 462 Gates Building, 353 Serra Mall, Stanford, CA 94305. Supported in part by ONR grant N00014-04-1-0725 and DARPA grant W911NF-04-9-0001. Most of this research was performed while the author was visiting NYU and supported by ONR grant N00014-98-1-0589 as a postdoctoral researcher at Cornell University. Email: tim@cs.stanford.edu.

- The largest-possible benefit from taxes does not exceed that from edge removals. In *every* network with linear latency functions, the benefit of taxes cannot exceed that of removing edges. There are networks with nonlinear latency functions, however, in which taxes are radically more powerful than edge removals.
- For every $\epsilon > 0$, there is no $(\frac{4}{3}-\epsilon)$-approximation algorithm for computing optimal taxes, even in networks with linear latency functions (assuming $P \neq NP$).

1 Introduction

1.1 Selfish Routing and Marginal Cost Pricing

We study economic incentives for influencing selfish behavior in networks. We focus on a simple model of *selfish routing*, defined by Wardrop [44] and first studied in the theoretical computer science literature by Roughgarden and Tardos [39]. In this model, we are given a directed network in which each edge possesses a latency function describing the common delay experienced by all traffic on the edge as a function of the edge congestion. There is a fixed amount of traffic that wishes to travel from a source vertex s to a sink vertex t; as in most earlier works, we assume that the traffic comprises a very large population of users, so that the actions of a single individual have negligible impact on the network congestion. The quality of an assignment of traffic to s-t paths is historically measured by the resulting sum of all travel times—the *total latency*. We assume that each network user, when left to its own devices, acts selfishly and routes itself on a minimum-latency path, given the network congestion caused by the other users. In general such a "selfish" assignment of traffic to paths (a *flow at Nash equilibrium*) does not minimize the total latency; put differently, the outcome of selfish behavior can be improved upon with coordination.

The inefficiency of selfish routing motivates the introduction of economic incentives to ensure that selfish behavior results in a socially desirable routing of traffic. An ancient idea—discussed informally as early as 1920 [35]—is to use *marginal cost pricing*. The principle of marginal cost pricing asserts that on each edge, each network user on the edge should pay a tax equal to the additional delay its presence causes for the other users on the edge. Assuming that all network users choose routes to minimize the sum of the latency experienced and the taxes paid, this principle ensures that the resulting flow at Nash equilibrium achieves the minimum-possible total latency [5]. Briefly, the inefficiency of selfish routing can always be eradicated by pricing network edges appropriately.

The following observation motivates our work: the principle of marginal cost pricing is single-minded in its pursuit of a minimum-latency flow, and ignores the disutility to network users due to (possibly very large) taxes. This assumption is only appropriate when collected taxes can be feasibly returned (directly or indirectly) to the network users, for example by refunding taxes equally to all users (a "lump-sum refund"). In this paper, we are interested in settings where this assumption is not reasonable. For example, refunding the collected taxes to network users could be logistically or economically infeasible, or taxes could represent quantities of a non-monetary, non-refundable good such as time delays.

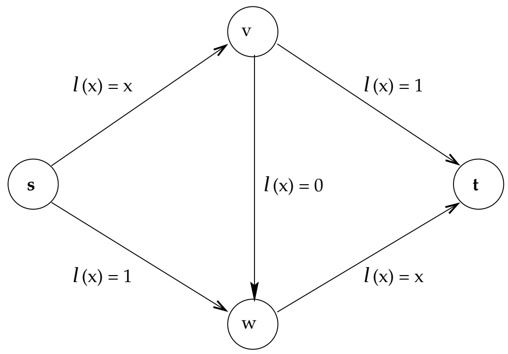


Figure 1: Braess's Paradox

If we wish to minimize the total user disutility—latency plus taxes paid, a quantity we call the *cost*—rather than merely the total latency, how should we price the network edges?

1.2 Braess's Paradox and the Power of Taxes

Intuition may suggest that taxes, which can only increase the disutility incurred by users along an s-t path, can never improve the cost of a flow at Nash equilibrium. We next describe *Braess's Paradox* [7], in a form first described by Schulman [40], which shows that this intuition is incorrect.

Each edge of the network in Figure 1 is labeled with its latency function, giving the delay incurred by traffic on the link as a function of the amount of traffic that uses the link. (We assume that there is one unit of traffic overall.) In the (unique) flow at Nash equilibrium, all traffic uses route the $s \to v \to w \to t$ and experiences two units of latency. On the other hand, if at least half a unit of tax is levied on the edge (v, w), then in the flow at Nash equilibrium half of the traffic uses each of the routes $s \to v \to t$ and $s \to w \to t$. In particular, the path $s \to v \to w \to t$ has latency 1 and cost at least 3/2 with respect to this flow, and hence does not offer an attractive alternative to traffic. In this new flow at Nash equilibrium, everyone experiences latency 3/2 and no taxes are paid. This outcome has cost 3/2 and is clearly superior to the original flow at Nash equilibrium in the absence of taxes.

For contrast, we next discuss the edge taxes that are dictated by the principle of marginal cost pricing for the network of Figure 1. As we will see in Subsection 3.1, these taxes are 1/2 on the edges (s, v) and (w, t) and 0 on the other three edges. With these taxes, we obtain the same flow at Nash equilibrium as in our previous solution with taxes, and all traffic experiences latency 3/2. However, all traffic must also pay 1/2 unit of tax. This solution thus has a cost of 2; evidently, the previous solution should be preferred over marginal cost taxes in this example.

The potential power of taxes to improve the cost of a flow at Nash equilibrium, together with the inadequacy of marginal cost pricing for this goal, motivate the questions that we study in this paper.

Section	Problem Studied	Linear	Arbitrary
3	Do marginal cost taxes ever help?	no	yes
4	Maximum benefit of taxes	4/3	$n/2$
5	Are taxes more powerful than edge removals?	no	yes
6	Approximability of optimal taxes	4/3	$[4/3, n/2]$

Table 1: Contributions of this paper. All results depend on the set of allowable edge latency functions. For simplicity, we list only results for networks with linear or arbitrary latency functions. The benefit of taxes is measured by the ratio in Nash flow cost before and after taxes are levied.

(1) Are marginal cost taxes ever a good idea for minimizing the cost of a flow at Nash equilibrium?

(2) A sufficiently large edge tax effectively removes the edge from the network, and the power of removing edges to improve a flow at Nash equilibrium is well understood [36]. Taxes are thus at least as powerful as edge removals; when are they strictly more powerful?

(3) Can we compute or approximate optimal taxes efficiently?

1.3 Our Results

Our contributions on these three questions are proofs of the following results.

- In every network with linear latency functions, marginal cost taxes do not improve the cost of a flow at Nash equilibrium.

- The maximum-possible benefit of taxes is no more than that of edge removals.

- For *every* network with linear latency functions—not merely worst-case examples—taxes cannot decrease the cost of a flow at Nash equilibrium beyond what can be achieved by removing edges. By contrast, there are networks with nonlinear latency functions in which taxes can radically improve over the best subgraph solution.

- For every $\epsilon > 0$, there is no $(\frac{4}{3} - \epsilon)$-approximation algorithm for computing optimal taxes, even in networks with linear latency functions (assuming $P \neq NP$). For networks with linear latency functions, this hardness result is optimal.

Table 1 summarizes our results. Determining the maximum-possible benefit of taxes and the (in)approximability of the problem of computing optimal taxes require only reasonably straightforward extensions of existing work on network design [36]; all other results of this paper require new constructions and proof approaches.

1.4 Related Work

As we have noted, marginal cost pricing in selfish routing networks was first proposed by Pigou [35]. The model of selfish routing studied in this paper was first mathematically formalized in the 1950s by Wardrop [44] and Beckmann, McGuire, and Winsten [5], and has been extensively studied ever since. Further discussion and many more references on both selfish routing and related network models can be found in [38].

We make no attempt to survey the vast literature on the optimal pricing of shared resources, and mention only a few references that can serve as a starting point for further reading. Surveys and recent work on pricing selfish routing networks by the transportation science community include [4, 6, 13, 14, 16, 22, 23, 41]. There have also been several recent theoretical computer science papers on the topic [9, 17, 18, 24]. While researchers have long realized that marginal cost taxes may cause users to pay more than is necessary (see e.g. [6, 22]), none of the above papers incorporated the taxes paid into the definition of social welfare as in the present work. Beyond selfish routing networks, there has been an enormous amount of research on pricing in various network models; see [1, 8, 20, 26, 30, 31] and the references therein for many examples.

We note that much of the aforementioned work can be interpreted in the context of general microeconomic theory; see the survey by Mirrlees [33], for example, for discussion and references on this point. We are not, however, aware of any work in the economics literature that directly applies to the questions posed in this paper.

Some of the issues that we study are similar in spirit to the work on "frugal mechanisms" pioneered by Archer and Tardos [2, 3] and studied further in [15, 25, 32, 42]. These papers seek mechanisms (such as auctions) that solve an optimization problem in an incentive-compatible way, but also make use of only moderate incentives.

The paper closest to the present work is that of Roughgarden [36] on designing networks for selfish users. The central questions of [36] concern the maximum-possible benefit from and the algorithmic complexity of removing edges from a network with selfish routing. These questions can be viewed as special cases of some of the problems considered here, with edge taxes restricted to be either 0 or $+\infty$. This paper extends some of the results of [36] to the setting of more general taxes, and in addition tackles problems that have not been previously considered.

1.5 Organization

In Section 2 we formally define our traffic routing model and review useful results from past works. In Section 3 we ask if marginal cost pricing can improve the cost of selfish routing, and we resolve this question in the negative for networks with linear latency functions. In Sections 4 and 5 we ask if the power of taxes exceeds that of edge removals. In Section 4 we adapt previous work on network design [36] to show that the largest-possible decrease in cost due to taxes cannot exceed that due to edge removals. In Section 5 we show that taxes never improve over the best solution achievable by removing edges in networks with linear latency functions, but can be radically more powerful than edge removals in networks with arbitrary latency functions. Section 6 studies the algorithmic problem of computing taxes to minimize the cost of a flow at Nash equilibrium. Section 7 concludes with several suggestions for future research.

2 Preliminaries

2.1 Networks and Flows

We follow the notation and conventions of Roughgarden and Tardos [39]. We study a single-commodity flow network, described by a directed graph $G = (V, E)$ with a source vertex s and a sink vertex t. We allow parallel edges but have no use for self-loops. We denote the set of simple s-t paths by $\mathcal{P}$, and we assume that this set is nonempty. A *flow* f is a nonnegative vector, indexed by $\mathcal{P}$. For a fixed flow f we define $f_e = \sum_{P\in\mathcal{P}:e\in P} f_P$ as the amount of traffic using edge e en route from s to t. With respect to a finite and positive *traffic rate* r, a flow f is said to be *feasible* if $\sum_{P\in\mathcal{P}} f_P = r$.

The network G suffers from congestion effects; to model this, we assume that each edge e possesses a nonnegative, continuous, nondecreasing *latency function* ℓ_e that describes the delay incurred by traffic on e as a function of the edge congestion f_e. The latency of a path P in G with respect to a flow f is then given by $\ell_P(f) = \sum_{e\in P} \ell_e(f_e)$. The quality of a flow is historically measured by its *total latency* $L(f)$, defined by

$$L(f) \equiv \sum_{P\in\mathcal{P}} \ell_P(f) f_P = \sum_{e\in E} \ell_e(f_e) f_e,$$

where the equality follows by writing $\ell_P(f)$ as a sum over edges and reversing the order of summation. We will call a flow minimizing $L(\cdot)$ *optimal* or *minimum-latency*. An optimal flow always exists, as the space of all flows is compact and $L(\cdot)$ is a continuous function.

Finally, we allow a set of nonnegative *taxes* $\{\tau_e\}_{e\in E}$ to be placed on the edges of a network G. We write $\tau_P = \sum_{e\in P} \tau_e$ for the total taxes on a path P. The *cost* $C(f, \tau)$ of a flow f in a network with taxes τ is the total disutility caused to network users, accounting for disutility due to both latency and taxes:

$$C(f, \tau) \equiv \sum_{P\in\mathcal{P}} [\ell_P(f) + \tau_P] f_P = \sum_{e\in E} [\ell_e(f_e) + \tau_e] f_e.$$

The functions $L(\cdot)$ and $C(\cdot, \tau)$ coincide if $\tau = 0$.[1] We call a triple (G, r, ℓ) an *instance*, and use the notation $(G, r, \ell + \tau)$ to denote an instance in which taxes τ have been levied on the edges of G.

2.2 Flows at Nash Equilibrium

We assume that noncooperative behavior results in a Nash equilibrium—a "stable point" in which no traffic has an incentive to unilaterally alter its strategy (i.e., its route from s to t).

[1] In several previous papers on selfish routing (without taxes), such as [37, 39], the terms total latency and cost were used synonymously.

We also assume that all agents seek to minimize the sum of the latency experienced and the tax paid.[2] We therefore expect that, in a flow at Nash equilibrium, all traffic is routed on paths with minimum-possible latency plus tax. Formally, we have the following definition.

Definition 2.1 A flow f feasible for $(G, r, \ell + \tau)$ is *at Nash equilibrium* or is a *Nash flow* if for all $P_1, P_2 \in \mathcal{P}$ with $f_{P_1} > 0$,

$$\ell_{P_1}(f) + \tau_{P_1} \leq \ell_{P_2}(f) + \tau_{P_2}.$$

We next discuss several useful properties of flows at Nash equilibrium in single-commodity networks. None of these results are new; for proofs, see the original research papers or the book by Roughgarden [38].

First is an alternative definition of a Nash flow, which is a simple consequence of the fact that such a flow routes traffic only on shortest paths with respect to latencies plus taxes.

Proposition 2.2 ([36]) *Let f be a flow feasible for $(G,\ r,\ \ell + \tau)$. For a vertex v in G, let $d(v)$ denote the length, with respect to edge lengths $\ell_e(f_e) + \tau_e$, of a shortest s-v path in G. Then*

$$d(w) - d(v) \leq \ell_e(f_e) + \tau_e$$

for all edges $e = (v, w)$, and f is at Nash equilibrium if and only if equality holds whenever $f_e > 0$.

An immediate consequence of Proposition 2.2 is that the property of being at Nash equilibrium is a property only of the flow vector on edges $\{f_e\}$ induced by a flow f, rather than on the particular path decomposition.

Corollary 2.3 *Suppose $f, \tilde{f}$ are flows for $(G, r, \ell + \tau)$ with $f_e = \tilde{f}_e$ for all edges e. Then f is at Nash equilibrium if and only if $\tilde{f}$ is at Nash equilibrium.*

We next discuss the existence and uniqueness of Nash flows.

Proposition 2.4 ([5, 12]) *Every instance $(G, r, \ell + \tau)$ admits a directed acyclic flow at Nash equilibrium.*

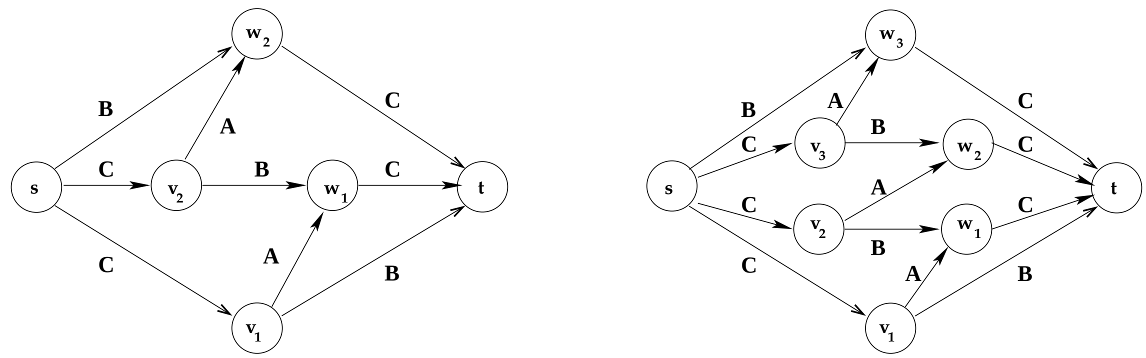
Proposition 2.5 ([5, 12]) *If $f, \tilde{f}$ are flows at Nash equilibrium for $(G, r, \ell + \tau)$, then:*

(a) $\ell_e(f_e) = \ell_e(\tilde{f}_e)$ for all edges e;

(b) $C(f, \tau) = C(\tilde{f}, \tau)$.

The next proposition states that all paths used by a Nash flow have the same combined latency and tax, and that the cost of a Nash flow is therefore expressible in a very simple form. It follows easily from Definition 2.1 and the definition of the cost a flow.

[2] This assumption, while classical, is obviously quite strong. In general, we expect different agents to have different objective functions and to trade off time and money in different ways. This objection raises several interesting issues that were studied in a recent sequence of papers [9, 17, 18, 24].



(a) B^2 (b) B^3

Figure 2: The second and third Braess graphs. Edges are labeled with their type.

Proposition 2.6 ([44]) *Let f be at Nash equilibrium for the instance $(G, r, \ell + \tau)$. Then, there is a constant $c \geq 0$ such that:*

(a) $\ell_P(f) + \tau_P = c$ whenever $f_P > 0$;

(b) $C(f, \tau) = r \cdot c$.

Propositions 2.5(b) and 2.6(b) imply that the constant c in Proposition 2.6 is independent of the chosen Nash flow f for $(G, r, \ell + \tau)$. We can therefore adopt the notation $c(G, r, \ell + \tau)$ for the value of this constant for the instance $(G, r, \ell + \tau)$.

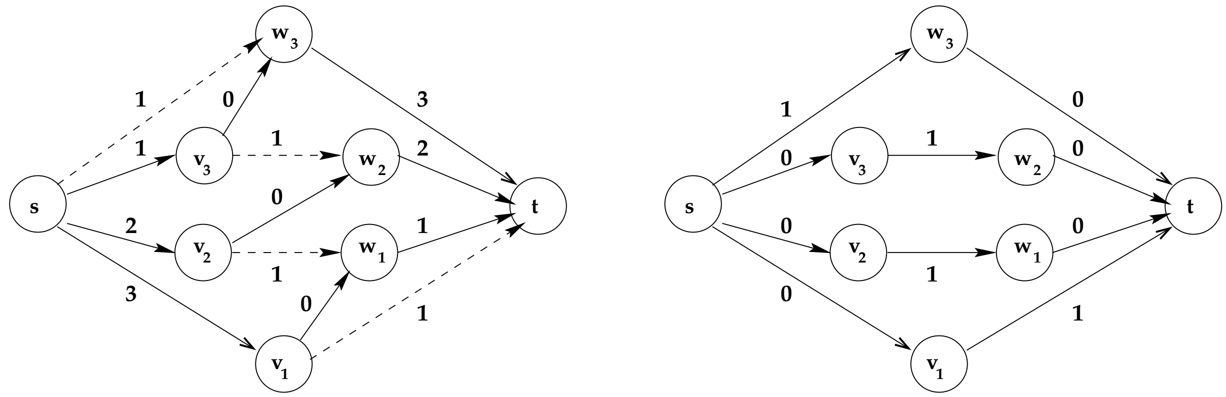
Our final proposition states that, for fixed G, ℓ, and τ, the value of $c(G, r, \ell + \tau)$ is nondecreasing in r. This fact was first proved by Hall [21] and we will use it in Section 3. For a combinatorial proof of the proposition, see Lin, Roughgarden, and Tardos [28].

Proposition 2.7 ([21]) *The value $c(G, r, \ell + \tau)$ is nondecreasing in r.*

2.3 The Braess Graphs

In this subsection we review the "Braess graphs". These networks were first defined by Roughgarden [36] to show that removing edges from a network can decrease the total latency of a Nash flow by an unbounded amount. As levying taxes can be viewed as a natural generalization of removing edges, these networks will also play an important role in this paper.

The *kth Braess graph* $B^k = (V^k, E^k)$ is defined as follows: start with a set $V^k = \{s, v_1, \ldots, v_k, w_1, \ldots, w_k, t\}$ of $2k + 2$ vertices and define E^k by $\{(s, v_i), (v_i, w_i), (w_i, t) : 1 \leq i \leq k\} \cup \{(v_i, w_{i-1}) : 2 \leq i \leq k\} \cup \{(v_1, t), (s, w_k)\}$ (see Figure 2). The graph B^1 is the same as Braess's original example (Figure 1).



(a) Nash flow for $(B^3, 3, \ell^3)$

(b) Nash flow in the optimal subgraph

Figure 3: Proof of Proposition 2.8, when $k = 3$. Solid edges carry traffic in the flow at Nash equilibrium, dashed edges do not. Edge latencies are with respect to flows at Nash equilibrium.

To describe latency functions for and flows in B^k in a convenient way, we introduce additional terminology. We call edges of the form (v_i, w_i) *type A edges*; edges of the form (v_i, w_{i-1}), (v_1, t), or (s, w_k) *type B edges*; and edges of the form (s, v_i) or (w_i, t) *type C edges* (see Figure 2). For $i = 1, \ldots, k$, let P_i denote the path $s \to v_i \to w_i \to t$. Finally, for $i = 2, \ldots, k$, let Q_i denote the path $s \to v_i \to w_{i-1} \to t$; define Q_1 to be the path $s \to v_1 \to t$ and Q_{k+1} the path $s \to w_k \to t$.

Braess graphs show that the latency of a Nash flow in an n-vertex network can be improved by an $\lfloor n/2 \rfloor$ factor by removing $\lfloor n/2 \rfloor - 1$ edges from the network.

Proposition 2.8 ([36]) *For every $n \geq 2$, removing edges from a network with n vertices and arbitrary latency functions can decrease the total latency of a Nash flow by a factor of $\lfloor n/2 \rfloor$.*

Proof: Fix n, which we can assume is at least 4, and let $k = \lfloor n/2 \rfloor - 1$. Define latency functions ℓ^k on the edges of B^k as follows: type A edges have constant latency 0; type B edges have constant latency 1; and for each $i \in \{1, 2, \ldots, k\}$, the type C edges (w_i, t) and (s, v_{k-i+1}) receive a continuous, nondecreasing latency function ℓ_e^k satisfying $\ell_e^k(k/(k+1)) = 0$ and $\ell_e^k(1) = i$.

Routing one unit of flow on each of the paths $P_1, \ldots, P_k$ gives a flow at Nash equilibrium for the instance (B^k, k, ℓ^k) in which all traffic experiences latency $k + 1$ (see Figure 3(a)). Removing the type A edges from B^k produces a subgraph H, and routing $k/(k + 1)$ units of flow on each of the paths $Q_1, \ldots, Q_{k+1}$ yields a flow at Nash equilibrium for (H, k, ℓ^k) with all traffic incurring latency 1 (see Figure 3(b)). ■

We will also study networks with edge latency functions that are degree-bounded polynomials with nonnegative coefficients. The following analogue of Proposition 2.8 holds for such networks.

Corollary 2.9 ([36]) *Removing edges from a network with latency functions that are polynomials with degree at most p and nonnegative coefficients can decrease the total latency of a Nash flow by an $\Omega(p/\log p)$ factor as $p \to \infty$.*

Corollary 2.9 can be proved with a modification of the construction used in Proposition 2.8, using the kth Braess graph with $k \approx p/\log p$. The traffic rate and latency functions are identical to those in the proof of Proposition 2.8, except that a type C edge of the form (w_i, t) or (s, v_{k-i+1}) receives the latency function ix^p.

Finally, a lower bound on the benefit achievable with edge removals in networks with linear latency functions follows from the network in Figure 1.

Corollary 2.10 ([7, 40]) *Removing edges from a network with linear latency functions can decrease the total latency of a Nash flow by a factor of $4/3$.*

3 When Do Marginal Cost Taxes Help?

In this section, we study the cost of applying the principle of marginal cost pricing. In Subsection 3.1 we formalize marginal cost taxes and the classical guarantee that they induce the minimum-latency flow as a flow at Nash equilibrium. In Subsection 3.2, however, we show that marginal cost taxes cannot decrease the cost of a Nash flow in a network with linear latency functions.

3.1 Marginal Cost Taxes Minimize Latency

Recall that the principle of marginal cost pricing posits that each user should pay a tax equal to the additional delay other users experience because of its presence. Mathematically, this principle asserts that in a flow f feasible for the instance (G, r, ℓ), the tax τ_e assigned to edge e should be $\tau_e = f_e \cdot \ell'_e(f_e)$, where ℓ'_e denotes the derivative of ℓ_e. (Assume for simplicity that the latency functions are differentiable.) The term $\ell'_e(f_e)$ corresponds to the marginal increase in latency caused by one user on the edge, and the term f_e is the amount of traffic that suffers from this increase. Marginal cost taxes come with the following guarantee, which is classical (see [5, 12, 38] for proofs).

Proposition 3.1 *Let (G, r, ℓ) be an instance with differentiable latency functions, admitting a minimum-latency flow f^*. Let $\tau_e = f^*_e \cdot \ell'_e(f^*_e)$ denote the marginal cost tax for edge e with respect to f^*. Then f^* is at Nash equilibrium for $(G, r, \ell + \tau)$.*

In words, marginal cost taxes induce the minimum-latency flow as a flow at Nash equilibrium.

3.2 Marginal Cost Taxes Increase Cost

Proposition 3.1 shows how to minimize the total latency of a Nash flow with edge taxes. But how effective are marginal cost taxes when we also account for the disutility to traffic due to taxes? We have already seen, in Subsection 1.2, that marginal cost taxes need not minimize the cost of a flow. Our next theorem identifies a reasonably large class of networks—networks in which all latency functions are linear, with the form $\ell(x) = ax + b$—in which marginal cost taxes are *guaranteed* to be unnecessary, if not detrimental. This result illustrates the dangers of marginal cost pricing when minimizing latency is not the sole goal.

Theorem 3.2 *Let f^* and f be minimum-latency and Nash flows, respectively, for an instance (G, r, ℓ) with linear latency functions. Let τ denote the marginal cost taxes with respect to f^*. Then,*

$$C(f, 0) \leq C(f^*, \tau).$$

Proof: Let (G, r, ℓ) be an instance with linear latency functions, with $\ell_e(x) = a_e x + b_e$ for each edge e (with $a_e, b_e \geq 0$). Let f^* and f be minimum-latency and Nash flows, respectively, for (G, r, ℓ). The principle of marginal cost pricing dictates that $\tau_e = f_e^* \cdot \ell_e'(f_e^*) = a_e f_e^*$ for each edge e.

Define the modified latency function ℓ_e^* by $\ell_e^*(x) = 2a_e x + b_e$. The functions $\ell + \tau$ and ℓ^* are not identically equal, but the identity

$$\ell_e^*(f_e^*) = 2a_e f_e^* + b_e = \ell_e(f_e^*) + \tau_e$$

holds for every edge e. Proposition 3.1 thus guarantees that f^* is at Nash equilibrium not only for the instance $(G, r, \ell + \tau)$, but also for the instance (G, r, ℓ^*). Moreover, in the notation of Proposition 2.6,

$$c(G, r, \ell^*) = c(G, r, \ell + \tau). \qquad (1)$$

We next claim that $f/2$ is at Nash equilibrium for $(G, r/2, \ell^*)$ with

$$c(G, r/2, \ell^*) = c(G, r, \ell). \qquad (2)$$

To see why, we note that since $\ell_e(x) = a_e x + b_e$ and $\ell_e^*(x) = 2a_e x + b_e$, edge and path latencies with respect to $f/2$ in $(G, r/2, \ell^*)$ and with respect to f in (G, r, ℓ) are identical. That f is a Nash flow for (G, r, ℓ) then implies that $f/2$ is at Nash equilibrium for $(G, r/2, \ell^*)$, with $c(G, r/2, \ell^*) = c(G, r, \ell)$.[3]

Combining (1) and (2) with Proposition 2.7, we obtain

$$c(G, r, \ell) = c(G, r/2, \ell^*) \leq c(G, r, \ell^*) = c(G, r, \ell + \tau).$$

The theorem now follows immediately from Proposition 2.6. ■

Remark 3.3 Theorem 3.2 concerns only networks with linear latency functions, but it can easily be extended to networks in which, for some fixed $p \geq 0$, every edge e has a latency function of the form $a_e x^p + b_e$ with $a_e, b_e \geq 0$.

[3] The essence of this proof first appeared in [39].

Remark 3.4 There are networks with nonlinear latency functions in which marginal cost taxes can decrease the cost of a Nash flow. For example, it is possible to define such an example using the network in Figure 1. The latency functions employed in this example are non-convex, and are similar to step functions. The details are somewhat tedious and we omit them. We do not view this example as a positive result for marginal cost taxes (as the example is contrived), but rather as justification for restricting the network latency functions in Theorem 3.2. We leave open the question of whether the negative result of Theorem 3.2 can be proved under significantly weaker assumptions on the latency functions.

4 How Powerful Are Arbitrary Taxes?

In this section we study the following question: how much can the cost of a Nash flow decrease after levying taxes on the edges? As we will see, a precise answer to this question follows easily from previous work on the power of edge removals [36].

The maximum-possible benefit from taxes will depend crucially on the allowable network latency functions. This dependence is characteristic of much of the work on selfish routing (see e.g. [38]). Indeed, we have already seen a glimpse of such a dependence in the previous section, where marginal cost pricing can decrease the cost of a Nash flow, but not in networks with linear latency functions.

Our first two upper bounds on the maximum-possible reduction in cost due to taxes are consequences of previous work on the *price of anarchy* [27, 34]. The price of anarchy of a set of selfish routing instances is the largest ratio between the total latency of a Nash flow of an instance and that of a minimum-latency flow for the instance. The price of anarchy is a function of the set of allowable latency functions, and this dependence is by now well understood. For example, the following statements are known (see [10, 37] for further examples).

Proposition 4.1 ([39]) *The price of anarchy in networks with linear latency functions is* $4/3$.

Proposition 4.2 ([37]) *The price of anarchy in networks with latency functions that are polynomials with degree at most* p *and nonnegative coefficients is asymptotically* $\Theta(p/\log p)$ *as* $p \to \infty$.

Upper bounds on the price of anarchy directly translate to upper bounds on the largest decrease in cost achievable with taxes: at best, taxes replace the Nash flow in the original network with the minimum-latency flow, while causing no additional disutility to network users. We therefore have the following corollaries.

Corollary 4.3 *Let* (G, r, ℓ) *be an instance with linear latency functions and* τ *a tax on edges. Let* f *and* f^τ *be Nash flows for* (G, r, ℓ) *and* $(G, r, \ell + \tau)$*, respectively. Then*

$$L(f) \leq \frac{4}{3} \cdot C(f^\tau, \tau).$$

Corollary 4.4 *There is a constant $c_1 > 0$ such that the following statement holds for all $p \geq 2$. If (G, r, ℓ) is an instance with polynomial latency functions with degree at most p and nonnegative coefficients, τ is a tax on edges, and f and f^τ are Nash flows for (G, r, ℓ) and $(G, r, \ell + \tau)$, respectively, then*

$$L(f) \leq c_1 \frac{p}{\log p} \cdot C(f^\tau, \tau).$$

In Subsection 2.3 we reviewed the Braess graphs, which give lower bounds on how much deleting edges can decrease the total latency of a Nash flow. As we have noted, sufficiently large taxes can simulate edge deletions, so these lower bounds carry over to the present setting. In particular, Corollary 2.10 implies that the bound of Corollary 4.3 is the best possible, and Corollary 2.9 demonstrates that Corollary 4.4 is optimal up to a constant factor.

In networks with arbitrary latency functions, the price of anarchy is unbounded, even in two-node, two-link networks [39]. While this might suggest that no finite upper bound on the largest-possible benefit of taxes is possible in such networks, a bounded price of anarchy is only a *sufficient* (and not necessary) condition for such a bound. Indeed, with no assumptions whatsoever on the network latency functions, we can still obtain an upper bound that is a function of the network size.

Theorem 4.5 *Let (G, r, ℓ) be an instance with n vertices and τ a tax on edges. Let f and f^τ be Nash flows for (G, r, ℓ) and $(G, r, \ell + \tau)$, respectively. Then*

$$L(f) \leq \left\lfloor \frac{n}{2} \right\rfloor \cdot C(f^\tau, \tau).$$

The proof of Theorem 4.5 is a straightforward extension of an argument from [36, Theorem 4.1], which proves the weaker statement that deleting edges from a network can reduce the total latency of a Nash flow by at most an $\lfloor n/2 \rfloor$ factor. We give the proof in Appendix A for completeness. Proposition 2.8 implies that the bound of Theorem 4.5 is the best possible.

5 Are Taxes More Powerful Than Edge Removals?

In Section 4, we saw that there is a strong connection between the power of taxes and the power of edge removals. Specifically, we found that for several natural classes of networks, the maximum-possible reduction in cost achievable by levying taxes on edges is the same as that by removing edges from the network. However, we have not resolved whether or not there exist *any* networks in which taxes can improve upon the best solution obtainable by removing edges. In other words, is the power of taxes no greater than that of edge removals even on an instance-by-instance basis? We study this question in this section.

In Subsection 5.1 we show that the answer is "yes" in networks with linear latency functions: in *every* such network, taxes cannot decrease the cost of a Nash flow more than edge removals can. By contrast, in Subsection 5.2 we show that in general networks, taxes can reduce the cost of a Nash flow far beyond what is achievable by merely deleting edges from the network.

5.1 The Power of Edge Removals in Networks with Linear Latency Functions

5.1.1 Overview

In this subsection we consider only networks with linear latency functions. Our main result is that taxes are never more powerful than edge removals in these networks. To state this formally, we will say that a set τ of taxes for the instance (G, r, ℓ) is $0/\infty$ if, for some Nash flow f^τ for $(G, r, \ell + \tau)$, $\tau_e = 0$ or $f^\tau_e = 0$ for each edge e. We note that $0/\infty$ taxes are no more powerful than edge removals, since if τ is $0/\infty$ then $c(G, r, \ell + \tau) = c(H, r, \ell)$, where H is the subgraph of G comprising the edges with zero tax.

We abuse notation and, with respect to an instance (G, r, ℓ), write $C(\tau)$ to denote $C(f^\tau, \tau)$, where f^τ is at Nash equilibrium for $(G, r, \ell + \tau)$. The function $C(\tau)$ is well defined by Proposition 2.5(b). A tax vector τ^* is *optimal* for an instance (G, r, ℓ) if $C(\tau^*) \leq C(\tau)$ for all nonnegative tax vectors τ. Because there are an infinite number of possible tax vectors, it is not even obvious that every instance admits an optimal set of taxes. The following result establishes the stronger statement that every instance (with linear latency functions) admits an optimal tax vector that is $0/\infty$.

Theorem 5.1 *An instance with linear latency functions admits an optimal set of taxes that is* $0/\infty$.

The proof of Theorem 5.1 is fairly involved. To avoid considering an arbitrary network with linear latency functions, we will argue by contradiction and study a minimal counterexample. As we will see, minimal counterexamples possess several convenient properties that facilitate the proof.

Precisely, a *counterexample* is an instance (G, r, ℓ) with linear latency functions that admits no optimal $0/\infty$ tax. A counterexample is *minimal* if no other counterexample has fewer edges. A tax τ is *good* for (G, r, ℓ) if $C(\tau) < C(\hat{\tau})$ for all $0/\infty$ taxes $\hat{\tau}$. If (G, r, ℓ) admits a good tax, then it is clearly a counterexample. We are not yet claiming that a counterexample (G, r, ℓ) admits an optimal tax; conceivably, the value $\inf\{C(\tau) : \tau \geq 0\}$ is not attained by any tax. However, since the set $\{C(\tau) : \tau \text{ is } 0/\infty\}$ is finite, every counterexample admits a good tax.

To achieve the desired contradiction, we will also need a tax that is in some sense minimal. Formally, we will call a tax *minimal* for an instance if it is an optimal tax and minimizes $\sum_{e \in E} \tau_e$ among all optimal taxes τ. Once we establish that minimal taxes exist for minimal counterexamples, the proof of Theorem 5.1 will be relatively short. Our key lemma is thus the following.

Lemma 5.2 *A minimal counterexample admits a minimal tax.*

5.1.2 Properties of Minimal Counterexamples

The proof of Lemma 5.2 makes use of several properties of minimal counterexamples. We establish these next.

First, minimal counterexamples are directed acyclic networks, with every good tax inducing a flow at Nash equilibrium that routes flow on every edge.

Lemma 5.3 *Let (G, r, ℓ) be a minimal counterexample, τ a good tax, and f^τ a flow at Nash equilibrium for $(G, r, \ell + \tau)$. Then $f^\tau_e > 0$ for all edges e of G.*

Proof: If $f^\tau_e = 0$ for some edge e of G, then deleting e from G yields a counterexample with fewer edges. ■

Corollary 5.4 *If (G, r, ℓ) is a minimal counterexample, then G is directed acyclic.*

Proof: The corollary follows immediately from Proposition 2.4 and Lemma 5.3. ■

Our next lemma states that Nash flows in minimal counterexamples are unique, up to the path decomposition of the induced flow on edges (cf., Corollary 2.3).

Lemma 5.5 *If (G, r, ℓ) is a minimal counterexample, τ is a tax, and f^τ is a Nash flow for $(G, r, \ell + \tau)$, then f is a Nash flow for $(G, r, \ell + \tau)$ if and only if $f_e = f^\tau_e$ for all edges e.*

Proof: The "if" direction is Corollary 2.3. For the "only if" direction, suppose for contradiction that f is a Nash flow for $(G, r, \ell + \tau)$ with $f_e \neq f^\tau_e$ for some edge e. Put $z_e = f_e - f^\tau_e$ for each edge e. Since f and f^τ are flows at the same traffic rate, z is a (signed) circulation: for each vertex v, with edges $\delta^+(v)$ having tail v and edges $\delta^-(v)$ having head v,

$$\sum_{e \in \delta^+(v)} z_e = \sum_{e \in \delta^-(v)} z_e.$$

By Proposition 2.5(a), z_e is non-zero only when e has a constant latency function.

Since (G, r, ℓ) is a counterexample, it admits a good tax $\hat{\tau}$. (Recall τ need not be good.) Let $f^{\hat{\tau}}$ be a Nash flow for $(G, r, \ell + \hat{\tau})$. Since $f^{\hat{\tau}}_e > 0$ for all edges e by Lemma 5.3, $\{f^{\hat{\tau}}_e + \lambda z_e\}_{e \in E}$ is a nonnegative vector for λ sufficiently near zero (positive or negative). In this case, it corresponds (after a path decomposition) to a flow f^λ feasible for $(G, r, \ell + \hat{\tau})$. Moreover, since z_e is non-zero only when e possesses a constant latency function, Proposition 2.2 implies that f^λ is at Nash equilibrium for $(G, r, \ell + \hat{\tau})$. Since we assumed that some z_e is non-zero, some choice of λ yields a Nash flow f^λ for $(G, r, \ell + \hat{\tau})$ with $f^\lambda_e = 0$ for some edge e. This contradicts Lemma 5.3. ■

Finally, we strengthen Proposition 2.6(a) for minimal counterexamples.

Lemma 5.6 *Let (G, r, ℓ) be a minimal counterexample, τ a good tax, and f^τ a Nash flow for $(G, r, \ell + \tau)$. Then there is a constant c such that*

$$\ell_P(f^\tau) + \tau_P = c$$

for every s-t path P of G.

Proof: By Lemma 5.3, we can choose a path decomposition f of $\{f^\tau_e\}_{e \in E}$ with $f_P > 0$ for all paths $P \in \mathcal{P}$. Corollary 2.3 implies that f is at Nash equilibrium for $(G, r, \ell + \tau)$; Proposition 2.6(a) implies that all s-t paths have a common latency plus tax. ■

5.1.3 Proof of Lemma 5.2

Recall that our key lemma, Lemma 5.2, asserts that a minimal counterexample admits a minimal tax. Our proof of this result requires two technical lemmas, which will allow us to prove the existence of optimal and minimal taxes for minimal counterexamples via compactness arguments. The first lemma states that bounded taxes suffice to minimize the cost of a Nash flow in a minimal counterexample.

Lemma 5.7 *Let (G, r, ℓ) be a minimal counterexample and τ a good tax. Let G have n vertices and define $\ell_{max} = \max_{e \in E} \ell_e(r)$. There is a tax $\tilde{\tau}$ with $\max_e \tilde{\tau}_e \leq n\ell_{max}$ and*

$$C(\tilde{\tau}) \leq C(\tau).$$

Proof: Let τ be a good tax for (G, r, ℓ), with f^τ at Nash equilibrium for $(G, r, \ell + \tau)$. We next show how to decrease taxes while leaving f^τ at Nash equilibrium.

By Corollary 5.4, G is directed acyclic and we can therefore order the vertices of G so that all edges of G travel forward. Since $f_e^\tau > 0$ on all edges of G (Lemma 5.3), s is the first vertex in the ordering and t is the last. Beginning with the penultimate vertex and proceeding backward in the ordering, we perform the following operation for each vertex $v \neq s$: let $\tau_v \geq 0$ denote the minimum tax on an edge with tail v, subtract τ_v from the tax of every edge with tail v and add τ_v to the tax of every edge with head v. This operation leaves the total tax of all s-t paths and the cost of all feasible flows unchanged. In particular, the flow f^τ remains at Nash equilibrium. When the source s is reached, subtract τ_s from the tax on all edges with tail s; f^τ remains at Nash equilibrium and its cost can only decrease. Call the new set of taxes $\tilde{\tau}$. We have already argued that $C(\tilde{\tau}) \leq C(\tau)$; it remains to show that $\max_e \tilde{\tau}_e \leq n\ell_{max}$.

We first observe that the tax-reducing operations iteratively enforce the following property: every vertex other than t has an outgoing edge with zero $\tilde{\tau}$-tax. This property implies that some s-t path, say P_0, has zero $\tilde{\tau}$-tax. Since $C(\tilde{\tau}) \leq C(\tau)$, $\tilde{\tau}$ is a good tax. Since f^τ is at Nash equilibrium for $(G, r, \ell + \tilde{\tau})$, Lemma 5.6 implies that

$$\ell_P(f^\tau) + \tilde{\tau}_P = \ell_{P_0}(f^\tau)$$

for every path $P \in \mathcal{P}$. Since $\ell_{P_0}(f^\tau) \leq n\ell_{max}$ and every edge of G lies on some s-t path (by Lemma 5.3), no edge tax in $\tilde{\tau}$ can exceed $n\ell_{max}$. The proof is complete. ■

The second technical lemma asserts continuity of the map $\tau \mapsto C(\tau)$.

Lemma 5.8 *Let (G, r, ℓ) be a minimal counterexample. Then the corresponding map $\tau \mapsto C(\tau)$ is continuous.*

Proof: Lemma 5.5 ensures that the map $\tau \mapsto \{f_e^\tau\}_{e \in E}$ is well defined. Lemma 5.5 and a result of Dafermos and Nagurney [11, Theorem 3.1] imply that this map is continuous. The lemma then follows easily. ■

We are finally prepared to prove Lemma 5.2.

Proof of Lemma 5.2: We first show that a minimal counterexample (G, r, ℓ) admits an optimal tax—that $\inf_{\tau \geq 0} C(\tau)$ is attained by some tax. Let B denote the taxes with all components bounded by $n\ell_{max}$, where n is the number of vertices of G and $\ell_{max} = \max_{e \in E} \ell_e(r)$. Since (G, r, ℓ) is a counterexample, the value $\inf_{\tau \geq 0} C(\tau)$ is approached by good taxes. Since (G, r, ℓ) is minimal, Lemma 5.7 implies that $\inf_{\tau \geq 0} C(\tau)$ is approached by good taxes in B:

$$\inf_{\tau \in B} C(\tau) = \inf_{\tau \geq 0} C(\tau).$$

Since B is a compact subset of $\mathcal{R}^E$ and C is continuous by Lemma 5.8, these infima are attained by some tax (in B).

Let O denote the (non-empty) set of optimal taxes for (G, r, ℓ). We will show that $\inf_{\tau \in O} \sum_e \tau_e$ is attained by some optimal tax. By Lemma 5.7, there is an optimal tax $\tau \in O$ with $\sum_{e \in E} \tau_e \leq mn\ell_{max}$, where m is the number of edges of G. Since O is the inverse image of a closed set under a continuous map (write $O = C^{-1}(C(\tau))$ for some $\tau \in O$), it is a closed subset of $\mathcal{R}^E$. Restricting O to taxes with sum of all components at most $mn\ell_{max}$, we obtain a nonempty compact subset $S \subseteq O$ of optimal taxes. Since $\tau \mapsto \sum_e \tau_e$ is a continuous function, it attains a minimum on S; this is also its minimum on O, and the proof is complete. ■

5.1.4 Proof of Theorem 5.1

With Lemma 5.2 in hand, we can now prove Theorem 5.1 by a perturbation argument.

Proof of Theorem 5.1: To derive a contradiction, let (G, r, ℓ) be a minimal counterexample, τ a minimal (and hence good) tax, and f^τ a Nash flow for $(G, r, \ell + \tau)$. Since (G, r, ℓ) is a counterexample, $\sum_e \tau_e > 0$. By Lemma 5.6, there is a constant c^τ such that $\ell_P(f^\tau) + \tau_P = c^\tau$ for all s-t paths P of G.

Write $\ell_e(x) = a_e x + b_e$ for each edge e of G. The equations

$$\sum_{e \in P} [a_e f_e + b_e + \tau_e] = c$$

for all $P \in \mathcal{P}$, together with the standard flow conservation constraints for f, form a system of equations linear in the $m + 1$ variables $\{f_e\}_{e \in E}$ and c (where m is the number of edges of G). By Lemma 5.5, $(\{f_e^\tau\}, c^\tau)$ is the unique solution to this system, with flow nonnegativity constraints automatically satisfied (indeed, strictly by Lemma 5.3). Choosing $m + 1$ linearly independent constraints with at least one constraint corresponding to a path with nonzero tax, there is a square linear system

$$A \begin{bmatrix} f \\ c \end{bmatrix} = d$$

for which $(\{f_e^\tau\}, c^\tau)$ is the unique solution, namely $A^{-1}d$.

Let $\tau_{\tilde{e}} > 0$ be a positive edge tax appearing in this linear system (as part of the right-hand side d). Consider perturbing this tax by subtracting a small number $\epsilon > 0$. This translates to a perturbation of adding ϵ to the right-hand side of all constraints corresponding to paths that include the edge $\tilde{e}$, resulting in the new right-hand side $\tilde{d}$. Since $f_e^\tau > 0$ for every edge e, the new Nash flow is given by

$$(\{\tilde{f}_e\}, \tilde{c}) = A^{-1}\tilde{d}$$

for sufficiently small perturbations. Since τ is an optimal tax minimizing $\sum_e \tau_e$, subtracting $\epsilon > 0$ from $\tau_{\tilde{e}}$ produces a non-optimal tax. By Proposition 2.6, it follows that $c^\tau < \tilde{c}$. By linearity, however, the opposite perturbation of adding ϵ to $\tau_{\tilde{e}}$ has the opposite effect, producing a tax $\bar{\tau}$ that induces a solution $(\{\bar{f}_e\}, \bar{c})$ with $\bar{c} < c^\tau$. This contradicts the optimality of τ, and the proof is complete. ■

5.2 The Power of Taxes in General Networks

The previous subsection showed that in every network with linear latency functions, taxes cannot improve over the best solution attainable by removing edges from the network. We now demonstrate that this result does not extend to networks with nonlinear latency functions. In fact, for each value of $n \geq 2$, there is an n-node network in which arbitrary nonnegative taxes can improve upon $0/\infty$ taxes by an $\lfloor n/2 \rfloor$ factor. With this result, we will have a good understanding of the relationship between taxes and edge removals in general networks. Briefly, removing edges can improve the cost of a Nash flow by an $\lfloor n/2 \rfloor$ factor (Proposition 2.8); taxes can improve the cost of a Nash flow by an $\lfloor n/2 \rfloor$ factor beyond what is achievable by removing edges (Theorem 5.9 below); but taxes (or edge removals) cannot improve the cost of a Nash flow by more than an $\lfloor n/2 \rfloor$ factor (Theorem 4.5).

Theorem 5.9 *For each integer $n \geq 2$, there is an instance (G, r, ℓ) with $c(H, r, \ell) \geq \lfloor n/2 \rfloor$ for all subgraphs H of G but $c(G, r, \ell + \tau) = 1$ for some tax $\tau \geq 0$.*

Proof: The construction is similar to that in an inapproximability result for network design [36, Theorem 4.3]. We can assume that n is even and at least 4. (For n odd, add an isolated vertex or subdivide an edge.) We will work with the Braess graph B^k for which $2k + 2 = n$. (See Subsection 2.3 for notation and terminology.)

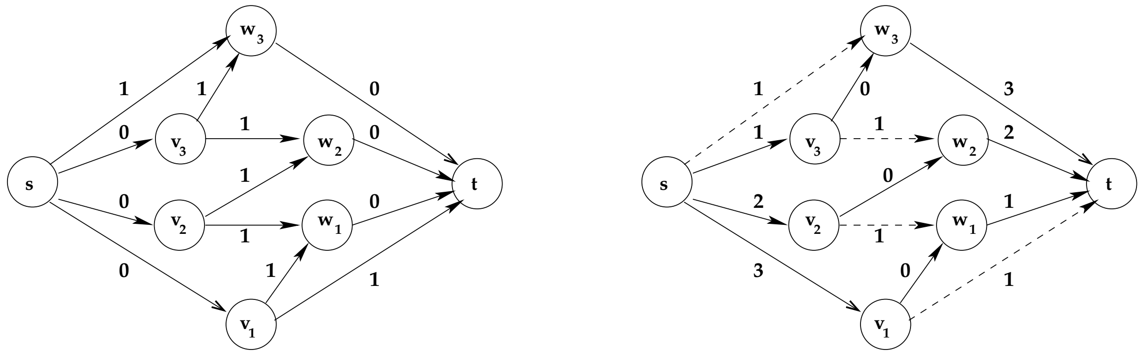
We define latency functions ℓ^k on the graph B^k as follows.

(A) Type A edges receive the latency function $\ell^k(x) = 0$.

(B) Type B edges are given a (continuous, nondecreasing) latency function ℓ^k satisfying $\ell^k(x) = 1$ for $x \leq 1/(k+1)$ and $\ell^k(x) = n/2$ for $x \geq 1/(k+1) + \epsilon$, where $\epsilon > 0$ is a sufficiently small constant.

(C) For each $i \in \{1, \ldots, k\}$, the type C edges (w_i, t) and (s, v_{k-i+1}) receive a latency function ℓ^k satisfying $\ell^k(x) = 0$ for $x \leq 1 + 1/(k+1)$, $\ell^k(1 + 1/k) = i$, and $\ell^k(x) = n/2$ for $x \geq 1 + 1/k + \epsilon$.

If a type B edge carries at least $1/(k+1) + \epsilon$ units of flow or a type C edge carries at least $1 + 1/k + \epsilon$ units of flow, we will say that the edge is *oversaturated*. A simple but important observation is that if a Nash flow oversaturates an edge in (H, r, ℓ^k) for some subgraph H of B^k, then $c(H, r, \ell^k) \geq n/2$.



(a) Latencies plus taxes in a Nash flow for $(B^k, k+1, \ell^k + \tau)$

(b) Latencies in a Nash flow for $(B^k, k+1, \ell^k)$

Figure 4: Proof of Theorem 5.9 when $k = 3$. Solid edges carry traffic in the flow at Nash equilibrium, dashed edges do not. Edges are labeled with their cost (sum of latency and tax), where latencies are with respect to flows at Nash equilibrium.

First, let τ be the tax vector equal to 1 on type A edges and 0 elsewhere. The following flow is then at Nash equilibrium for $(B^k, k+1, \ell^k + \tau)$: route 1 unit of flow on each of $P_1, P_2, \ldots, P_k$ and $1/(k+1)$ units of flow on each of $Q_1, Q_2, \ldots, Q_{k+1}$. Type A edges then each have zero latency and one unit of tax, type B edges each have zero tax and one unit of latency, and type C edges have zero latency and tax (see Figure 4(a)). This Nash flow proves that $c(B^k, k+1, \ell^k + \tau) = 1$.

To finish the proof, we need to show that $c(H, k+1, \ell^k) \geq n/2$ for every subgraph H of B^k; this requires a bit of case analysis. If H is all of B^k, then $c(H, k+1, \ell^k) = n/2$ because routing $1 + 1/k$ units of traffic on each of $P_1, P_2, \ldots, P_k$ provides a flow at Nash equilibrium (see Figure 4(b)). Similarly, $c(H, k+1, \ell^k) = n/2$ if H omits only type B edges.

Next, suppose H omits some type C edge and $\epsilon > 0$ is sufficiently small. If edge (s, v_i) is not in H, then every flow feasible for $(H, k+1, \ell^k)$ oversaturates some edge incident to s. Thus $c(H, k+1, \ell^k) \geq n/2$ if H omits a type C edge incident to s. A symmetric argument applies to subgraphs H that omit a type C edge incident to t.

Finally, suppose H omits some type A edge, say (v_i, w_i). The vertex v_i then has at most one outgoing edge in H, which must be a type B edge. If this edge is not oversaturated, then the edge (s, v_i) carries at most $\frac{1}{k+1} + \epsilon$ units of flow; as in the previous paragraph, this implies that some edge incident to s is oversaturated. In either case, $c(H, k+1, \ell^k) \geq n/2$ and the proof is complete. ■

6 The Complexity of Computing Optimal Taxes

In this section, we study the optimization problem of minimizing the cost of a Nash flow by taxing the network edges, and extend an existing hardness result for network design [36] to this problem.

By an *α-approximation algorithm* for a minimization problem, we mean an algorithm that runs in polynomial time and returns a solution no more than α times as costly as an optimal solution. We will call the value α the *approximation ratio* or *performance guarantee* of the algorithm.

The maximum-possible benefit achievable with taxes, as determined in Section 4, has immediate consequences for the performance guarantee of the *trivial algorithm*—the algorithm that assigns all edges zero tax. In particular, Corollary 4.3 implies the following.

Corollary 6.1 *The trivial algorithm is a $\frac{4}{3}$-approximation algorithm for the problem of taxing edges to minimize the cost of a Nash flow in networks with linear latency functions.*

Roughgarden [36] gave several inapproximability results for the problem of removing edges from a network to minimize the total latency of a Nash flow. We next extend one of them to the problem of computing optimal taxes in networks with linear latency functions.

Theorem 6.2 *For every $\epsilon > 0$, there is no $(\frac{4}{3} - \epsilon)$-approximation algorithm for the problem of taxing edges to minimize the cost of a Nash flow in networks with linear latency functions (unless $P = NP$).*

Corollary 6.1 and Theorem 6.2 imply that, in networks with linear latency functions, no polynomial-time algorithm has approximation ratio better than that of the trivial algorithm (assuming $P \neq NP$).

The reduction used to prove Theorem 6.2 is identical to one in [36], and is from a disjoint paths problem. However, the proof of Theorem 6.2 is slightly more involved than that of its analogue in [36], due to the extra power of taxes beyond that of edge removals. In particular, proving that "no" instances of a disjoint paths problem give instances in which all possible taxes induce a costly Nash flow is harder than showing that all possible subgraphs of these instances have costly Nash flows.

For completeness, we prove Theorem 6.2 in Appendix B.

Remark 6.3 For the problem of removing edges from a network to minimize the total latency of a Nash flow, Roughgarden [36] also gave inapproximability results for networks with different types of nonlinear latency functions. These showed that the trivial algorithm is an optimal approximation algorithm for the problem in these classes of networks. We believe that most (if not all) of these hardness results should carry over to the problem of computing optimal taxes in networks with nonlinear latency functions, but we have been unable to verify the details.

7 Directions for Further Research

We have undertaken the first study of using edge taxes to minimize the cost—latency plus taxes paid—of a Nash flow. While we have answered several basic questions, some obvious gaps in our results remain. We next list three of the more glaring ones.

(Q1) Suppose we consider restricted latency functions that need not be linear, such as convex functions or degree-bounded polynomials with nonnegative coefficients. Can marginal cost taxes improve the cost of a Nash flow? Can levying taxes decrease the cost of a Nash flow beyond what is achievable with edge removals?

(Q2) Is the trivial algorithm an optimal approximation algorithm for the problem of computing optimal taxes in networks with nonlinear latency functions?

(Q3) Which results of this paper remain true in multicommodity flow networks, with multiple sources and destinations? (See Lin et al. [29] for some very recent work on this question.)

A broader research issue is to study other approaches to simultaneously minimizing both the total latency and the taxes paid by traffic. For example, Hearn and Ramana [22] show how to efficiently compute, among all tax vectors that induce a minimum-latency flow as a flow at Nash equilibrium, the tax vector minimizing the amount of tax paid by the traffic. (Of course, even the best such tax vector might require traffic to pay exorbitant taxes.) A different objective is to minimize the total latency of traffic, subject to a fixed budget on the amount of taxes that can be paid. Are there non-trivial algorithmic results for this problem?

Acknowledgements

We thank five anonymous referees for their comments on earlier drafts of this paper, and David Parkes for suggesting that alternative formulations of the goal of minimizing both latency and taxes might elude the hardness result in Section 6.

References

[1] E. Altman, T. Boulogne, R. El Azouzi, T. Jiménez, and L. Wynter. A survey on networking games in telecommunications. *Computers & Operations Research*, 2005. To appear.

[2] A. Archer and É. Tardos. Truthful mechanisms for one-parameter agents. In *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 482–491, 2001.

[3] A. Archer and É. Tardos. Frugal path mechanisms. In *Proceedings of the 13th Annual Symposium on Discrete Algorithms*, pages 991–999, 2002. Full version to appear in *Journal of Algorithms*.

[4] R. Arnott and K. Small. The economics of traffic congestion. *American Scientist*, 82(5):446–455, 1994.

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

[6] P. Bergendorff, D. W. Hearn, and M. V. Ramana. Congestion toll pricing of traffic networks. In P. M. Pardalos, D. W. Hearn, and W. W. Hager, editors, *Network Optimization*, pages 51–71. Springer-Verlag, 1997.

[7] D. Braess. Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung*, 12:258–268, 1968.

[8] R. Cocchi, S. J. Shenker, D. Estrin, and L. Zhang. Pricing in computer networks: Motivation, formulation, and example. *IEEE/ACM Transactions on Networking*, 1(6):614–627, 1993.

[9] R. Cole, Y. Dodis, and T. Roughgarden. Pricing network edges for heterogeneous selfish users. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 521–530, 2003.

[10] J. R. Correa, A. S. Schulz, and N. E. Stier Moses. Selfish routing in capacitated networks. *Mathematics of Operations Research*, 29(4):961–976, 2004.

[11] S. C. Dafermos and A. Nagurney. Sensitivity analysis for the asymmetric network equilibrium problem. *Mathematical Programming*, 28(2):174–184, 1984.

[12] S. C. Dafermos and F. T. Sparrow. The traffic assignment problem for a general network. *Journal of Research of the National Bureau of Standards, Series B*, 73B(2):91–118, 1969.

[13] R. B. Dial. Network-optimized road pricing: Part I: A parable and a model. *Operations Research*, 47(1):54–64, 1999.

[14] R. B. Dial. Network-optimized road pricing: Part II: Algorithms and examples. *Operations Research*, 47(2):327–336, 1999.

[15] E. Elkind, A. Sahai, and K. Steiglitz. Frugality in path auctions. In *Proceedings of the 15th Annual Symposium on Discrete Algorithms (SODA)*, pages 694–702, 2004.

[16] P. Ferrari. Road network toll pricing and social welfare. *Transportation Research, Part B*, 36(5):471–483, 2002.

[17] L. K. Fleischer. Linear tolls suffice: New bounds and algorithms for tolls in single source networks. In *Proceedings of the 31st Annual International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 3142 of *Lecture Notes in Computer Science*, pages 544–554, 2004.

[18] L. K. Fleischer, K. Jain, and M. Mahdian. Tolls for heterogeneous selfish users in multicommodity networks and generalized congestion games. In *Proceedings of the 45th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 277–285, 2004.

[19] S. Fortune, J. E. Hopcroft, and J. Wyllie. The directed subgraph homeomorphism problem. *Theoretical Computer Science*, 10(2):111–121, 1980.

[20] R. J. Gibbens and F. P. Kelly. Resource pricing and the evolution of congestion control. *Automatica*, 35(12):1969–1985, 1999.

[21] M. A. Hall. Properties of the equilibrium state in transportation networks. *Transportation Science*, 12(3):208–216, 1978.

[22] D. W. Hearn and M. V. Ramana. Solving congestion toll pricing models. In P. Marcotte and S. Nguyen, editors, *Equilibrium and Advanced Transportation Modeling*, pages 109–124. Kluwer Academic Publishers, 1998.

[23] B. Johansson and L. Mattsson. *Road Pricing: Theory, Empirical Assessment and Policy*. Kluwer, 1995.

[24] G. Karakostas and S. G. Kolliopoulos. Edge pricing of multicommodity networks for heterogeneous selfish users. In *Proceedings of the 45th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 268–276, 2004.

[25] A. Karlin, D. Kempe, and T. Tamir. Beyond VCG: Frugality of truthful mechanisms. In *Proceedings of the 46th Annual Symposium on Foundations of Computer Science (FOCS)*, 2005. To appear.

[26] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan. Rate control for communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, 1998.

[27] E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 1563 of *Lecture Notes in Computer Science*, pages 404–413, 1999.

[28] H. Lin, T. Roughgarden, and É. Tardos. A stronger bound on Braess's Paradox. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 333–334, 2004.

[29] H. Lin, T. Roughgarden, É. Tardos, and A. Walkover. Braess's Paradox, Fibonacci numbers, and exponential inapproximability. In *Proceedings of the 32nd Annual International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 3580 of *Lecture Notes in Computer Science*, pages 497–512, 2005.

[30] J. K. MacKie-Mason and H. R. Varian. Pricing congestible network resources. *IEEE Journal on Selected Areas in Communications*, 13(7):1141–1149, 1995.

[31] L. W. McKnight and J. P. Bailey. *Internet Economics*. MIT Press, 1998.

[32] M. Mihail, C. H. Papadimitriou, and A. Saberi. On certain connectivity properties of the Internet topology. In *Proceedings of the 44th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 28–35, 2003.

[33] J. A. Mirrlees. The theory of optimal taxation. In K. J. Arrow and M. D. Intriligator, editors, *Handbook of Mathematical Economics, Volume III*, chapter 24, pages 1197–1249. North-Holland, 1986.

[34] C. H. Papadimitriou. Algorithms, games, and the Internet. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 749–753, 2001.

[35] A. C. Pigou. *The Economics of Welfare*. Macmillan, 1920.

[36] T. Roughgarden. Designing networks for selfish users is hard. In *Proceedings of the 42nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 472–481, 2001. Full version to appear in *Journal of Computer and System Sciences*.

[37] T. Roughgarden. The price of anarchy is independent of the network topology. *Journal of Computer and System Sciences*, 67(2):341–364, 2003. Preliminary version in *STOC '02*.

[38] T. Roughgarden. *Selfish Routing and the Price of Anarchy*. MIT Press, 2005.

[39] T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002. Preliminary version in *FOCS '00*.

[40] L. Schulman. Personal communication. October 1999.

[41] M. J. Smith. The marginal cost taxation of a transportation network. *Transportation Research, Part B*, 13(3):237–242, 1979.

[42] K. Talwar. The price of truth: Frugality in truthful mechanisms. In *Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 2607 of *Lecture Notes in Computer Science*, pages 608–619, 2003.

[43] R. E. Tarjan. *Data Structures and Network Algorithms*. SIAM, 1983.

[44] J. G. Wardrop. Some theoretical aspects of road traffic research. In *Proceedings of the Institute of Civil Engineers, Pt. II*, volume 1, pages 325–378, 1952.

A Proof of Theorem 4.5

Proof of Theorem 4.5: Let (G, r, ℓ) be an instance in which G has n vertices, and let τ be a tax on the edges. Let f be a directed acyclic Nash flow for (G, r, ℓ) (see Proposition 2.4), and f^* a Nash flow for $(G, r, \ell + \tau)$. Let d and d^* be the corresponding distance labels of

Proposition 2.2. In the notation of Proposition 2.6, we need to prove that $D \equiv d(t) = c(G, r, \ell)$ is at most $\lfloor n/2 \rfloor$ times $D^* \equiv d^*(t) = c(G, r, \ell + \tau)$.

An ordering of the vertices of G is *good* if it satisfies the following two properties.

(P1) All f-flow travels forward in the ordering.

(P2) The d-values of vertices are nondecreasing in the ordering.

There is at least one good ordering. To see why, first topologically sort the vertices of G according to the (directed acyclic) flow f to ensure property (P1). An ordered pair (u, v) of vertices is *bad* if $d(v) < d(u)$ in spite of v following u in the ordering. Property (P2) is equivalent to the absence of bad vertex pairs.

If there is a bad vertex pair, there is one such pair (v, w) with v and w adjacent in the ordering. We would like to transpose v and w. By Proposition 2.2 and the nonnegativity of latency functions, d-values cannot decrease across an edge e with $f_e > 0$. Hence, there is no flow-carrying edge from v to w. Transposing v and w therefore does not violate property (P1) and strictly decreases the number of bad vertex pairs. Finitely many such transpositions yields a good ordering.

Place a good ordering on the vertices of G and label them $v_0, v_1, \ldots, v_{n-1}$ accordingly. We can assume that $v_0 = s$. Call an edge e of G *light* if $f_e \leq f_e^*$ with $f_e^* > 0$, and *heavy* otherwise. We can finish the proof by establishing two claims (see also Figure 5).

(1) If v_j precedes t in the good ordering, then there is a path of light edges beginning in $\{v_0, v_1, \ldots, v_j\}$ and terminating in $\{t, v_{j+2}, v_{j+3}, \ldots, v_{n-1}\}$.

(2) If there is a path of light edges from u to v, then $d(v) \leq d(u) + D^*$.

Since $d(s) = 0$ and d-values are nondecreasing in the good ordering, applying these two claims inductively to the sets $\{v_0, \ldots, v_{2i}\}$ gives $d(v_{2i}) \leq i \cdot D^*$ for v_{2i} equal to or preceding t. If $t = v_{2i}$ for an integer i, the theorem follows immediately. If $t = v_{2i+1}$, then $d(v_{2i}) \leq i \cdot D^*$ and the theorem follows from one further application of the two claims (to $\{v_0, \ldots, v_{2i}\}$).

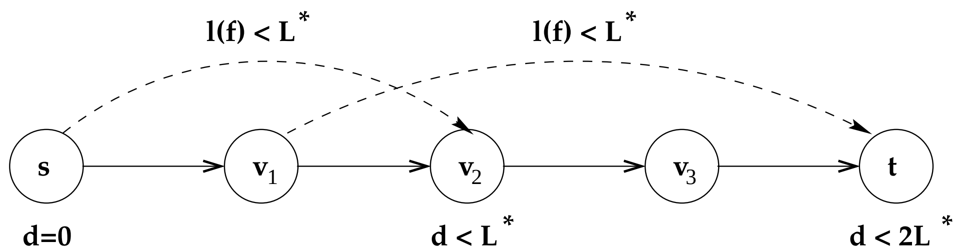


Figure 5: Proof of Theorem 4.5. If f is the flow sending one unit of flow on the four-hop path and f^* is the flow sending half a unit of flow on each of the other two paths, then the dashed edges are light.

To prove the first claim, let v_j precede t in the good ordering. By property (P1) of good orderings, no f-flow enters the s-t cut $S = \{v_0, \ldots, v_j\}$. Since the net f-flow and f^*-flow escaping any s-t cut is precisely r (see e.g. [43, Lemma 8.1]), at least one light edge escapes S. If some such edge has its head in $\{t, v_{j+2}, \ldots, v_{n-1}\}$, we are done. If not, all such light edges terminate at a vertex v_{j+1} that precedes t in the ordering. By the above argument, some light edge e escapes $\{v_0, \ldots, v_{j+1}\}$. Since all light edges emanating from S end at v_{j+1}, e begins at v_{j+1}. Thus, its concatenation with any light edge escaping S provides the desired path of light edges.

For the second claim, let P be a path of light edges from u to v. Since taxes are nonnegative and latency functions are nondecreasing, $\ell_e(f_e) \leq \ell_e(f_e^*) + \tau_e$ for every edge e in P. Since $f_e^* > 0$ for all edges e of P, Proposition 2.2 implies that

$$d(v) - d(u) \leq \sum_{e \in P} \ell_e(f_e) \leq \sum_{e \in P} [\ell_e(f_e^*) + \tau_e] = d^*(v) - d^*(u).$$

Similarly,

$$0 \leq d^*(u) \leq d^*(v) \leq D^*$$

and hence $d(v) - d(u) \leq d^*(v) - d^*(u) \leq D^*$. This completes the proof of the claim and hence the theorem. ■

B Proof of Theorem 6.2

Proof of Theorem 6.2: As in [36, Theorem 3.3], our reduction will proceed from the problem 2 Directed Disjoint Paths (2DDP): given a directed graph $G = (V, E)$ and distinct vertices $s_1, s_2, t_1, t_2 \in V$, are there s_i-t_i paths P_i for $i = 1, 2$ such that P_1 and P_2 are vertex-disjoint? Fortune, Hopcroft, and Wyllie [19] showed that this problem is NP-complete. We will show how a $(\frac{4}{3} - \epsilon)$-approximation algorithm for minimizing the cost of a Nash flow with taxes in a network with linear latency functions can be used to distinguish "yes" and "no" instances of 2DDP in polynomial time.

Let $\mathcal{I} = (G = (V, E), s_1, s_2, t_1, t_2)$ be an instance of 2DDP. Augment the vertex set V by a source s, a sink t, and directed edges (s, s_1), (s, s_2), (t_1, t), and (t_2, t) (see Figure 6). Denote the new network by $G' = (V', E')$. We give the edges of E' the following (linear) latency functions: all edges of E receive the latency function $\ell(x) = 0$, edges (s, s_2) and (t_1, t) are given the latency function $\ell(x) = x$, and edges (s, s_1) and (t_2, t) are endowed with the latency function $\ell(x) = 1$.

Following the notation of Subsection 5.1, we will write $C(\tau)$ for the cost of a Nash flow in $(G', 1, \ell + \tau)$. To complete the proof, we need only show the following two statements:

(i) if $\mathcal{I}$ is a "yes" instance of 2DDP, then there is a set τ of taxes with $C(\tau) = 3/2$;

(ii) if $\mathcal{I}$ is a "no" instance, then $C(\tau) \geq 2$ for all nonnegative taxes τ.

To prove (i), let P_1 and P_2 be vertex-disjoint s_1-t_1 and s_2-t_2 paths in G, respectively. Assign zero taxes to edges in $E' \setminus E$, P_1, and P_2. Set taxes τ_e on all other edges e to be sufficiently large. Then, routing half a unit of flow on each of P_1 and P_2 yields a flow at Nash equilibrium proving that $C(\tau) = 3/2$.

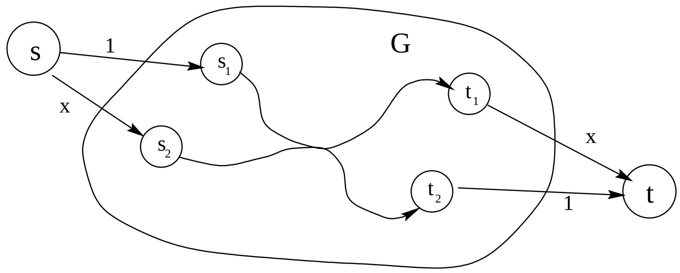


Figure 6: Proof of Theorem 6.2. In a "no" instance of 2DDP, s_1-t_1 and s_2-t_2 paths must share a vertex.

For (ii), consider taxes τ and a Nash flow f^τ for $(G', 1, \ell + \tau)$. Let d denote the distance labels of Proposition 2.2. Since there is one unit of traffic, (ii) is tantamount to showing that $d(t) \geq 2$.

First, if f^τ routes all flow through a single s_i-t_j pair, then the latency of all flow paths is 2 and hence $d(t) \geq 2$. We can thus assume that f^τ routes flow through at least two distinct s_i-t_j pairs. Similarly, we can assume that no flow path contains both s_1 and t_2.

Now suppose f^τ routes flow on a path P_1 containing s_1 and t_1 and also on a path P_2 containing s_2 and t_2. Since $\mathcal{I}$ is a "no" instance of 2DDP, P_1 and P_2 share an internal vertex v (see Figure 6). Since P_1 is a flow path of the Nash flow f^τ that contains v and the edge (s, s_1), Proposition 2.2 implies that $d(v) \geq 1$. Similarly, since P_2 is a flow path containing v and the edge (t_2, t), Proposition 2.2 implies that $d(t) \geq d(v) + 1 \geq 2$.

The final case arises when all flow paths contain a single s_i or a single t_i (but not both). If all flow paths contain a single s_i, then all flow uses edge (s, s_i), so $d(s_i) \geq 1$. As in the previous paragraph, since there is a flow path of f^τ containing s_i and edge (t_2, t), Proposition 2.2 implies that $d(t) \geq d(s_i) + 1 \geq 2$. A similar argument proves that $d(t) \geq 2$ if all flow paths of f^τ contain a single t_i. ■